INTERPLANETARY GUIDANCE SYSTEM REQUIREMENTS STUDY

VOLUME II

COMPUTER PROGRAM DESCRIPTIONS

PART 7

PERFORMANCE ASSESSMENT OF ATMOSPHERIC ENTRY GUIDANCE SYSTEMS

Prepared for
ERC Systems Laboratory
under Contract NAS 12-7

AC ELECTRONICS DIVISION
General Motors Corporation
El Segundo, California

31 August 1966

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

This page left blank intentionally

ABSTRACT

This document describes a digital computer program developed to simulate the performance assessment of atmospheric entry aided-inertial guidance systems. The description contains a mathematical model, a computer program description, a user's guide, an operator's and programmer's guide, and a program listing. The program is written in FORTRAN IV for the IBM 7094.

1.0 INTRODUCTION AND SUMMARY

This document describes in detail a digital computer program for the performance assessment of aided inertial atmospheric entry guidance systems. The description includes a statement of the basic mathematical assumptions and a description of the mathematical model based on these assumptions. The mathematical model assumes optimal data processing of IMU and electromagnetic sensor input in real time, the availability of a nominal trajectory, and the application of lambda matrix guidance. In combination with the nominal trajectory program as described in Volume II, Parts 5 and 6, the performance of such aided inertial atmospheric guidance systems can be evaluated for single pass, single skipout, and atmospheric braking trajectories.

The description of the mathematical model is followed by a representation of the computer program itself in terms of flow charts and equations in Section 3. The flow charts are intended to indicate a logical flow connecting different functional blocks. Although these flow charts may not always describe the literal operation within the program, nor do they include the myriad of small details consistent with a large complicated program, the blocks are numbered and these may be identified with the various subroutines by means of information appearing in the Operator's and Programmer's Guide. A listing of the program is also provided, together with a key which identifies the FORTRAN symbols in the program with the equation symbols in the flow charts.

The flow charts have been arranged and drawn according to a hierarchical structure. The "highest" level, designated as Level I, depicts the overall structure of the program. Each block appearing in this chart is described by another flow chart. These charts are designated as Level II. This policy is repeated for each block in every level until no further logic remains to be described. The final set of flow charts at the lowest level are supplemented by the detailed equations which are used in the program. Of particular interest to one inexperienced in the use of the program is the User's Guide. Instructions are included here which indicate how to supply all the necessary input in order to operate this program in any of its sixteen operational modes. The information contained in this section along with instructions to the operator contained in the Operator's and Programmer's Guide is adequate to operate this program to its fullest capability.

2.0 MATHEMATICAL MODEL

In this paragraph the basic equations necessary to specify the functions and assess the performance of a space guidance system during atmospheric entry phases of interplanetary missions are presented. The atmospheric entry may be accomplished on a single pass or a single skipout of the atmosphere may be utilized in order to extend the range capability. This program may also be used to study missions in which a space vehicle enters the atmosphere of a planet to accomplish atmospheric braking resulting in a change in the direction and magnitude of the vehicle's velocity vector.

2.1 GENERAL PERFORMANCE ASSESSMENT PROBLEM

The quantitative formulation of a mathematical model for the performance assessment of aided-inertial space guidance systems must be preceded by a definition of the term "space navigation and guidance system" and its function. In the context of this report the term "space navigation and guidance system" is defined as the entity of all on-board as well as ground-based equipment (where "ground" may not be identical with earth) necessary to perform the guidance and navigation functions. The functions of any guidance system are threefold: First, the state of the system (position, velocity, and attitude) must be determined. This mode is called navigation and consists of data gathering by the available sensors and the transmission, and processing of this data by airborne or ground-based equipment. Based upon this knowledge of the present state of the system, control actions can be computed according to a certain philosophy which will transfer the system from the present state into a desired state at a later time. This is the so-called guidance mode. Finally, the computed control actions are carried out in the control mode. This constitutes the third main function of the guidance system.

In order to give the problem a more definite character, the system configuration, which can perform these functions, must be specified in more detail. Two basically different types of sensors are, and will be, available for the navigation function, namely, inertial sensors and electromagnetic sensors. Inertial sensors measure nongravitational accelerations (accelerometers) such as engine-thrust and drag-forces and attitude or attitude rates (gyroscopes). The function of the inertial sensors is that of measuring the magnitude and direction of aerodynamic acceleration and, by comparison with nominal values, to improve the estimate of the state. This may be contrasted to the customary application of the IMU in which the state of the vehicle is calculated by propagrating the initial state. This is normally accomplished by measuring the nongravitational acceleration, calculating the gravitational acceleration and integrating these quantities to obtain position and velocity. The latter application does not permit one to improve ones knowledge of the initial conditions and in fact these initial errors are augmented since they contribute to errors in the gravity vector. Electromagnetic sensors are useful because their outputs are functions of the state of the vehicle only and because the output does not depend upon an integration process.

In addition the effectiveness of the electromagnetic sensors is undiminished, in contrast to that of the IMU, during the periods when aerodynamic forces are small with respect to gravitational forces.

The following electromagnetic sensors will be considered:

1. Three ground trackers providing range, range rate, and angular information
2. Horizon sensor measuring the planet's subtended angle and providing information about the local vertical
3. Radio altimeter measuring the altitude and radial speed of the vehicle

The performance of a specific space guidance system configuration will be measured in terms of the terminal accuracy and the control effort necessary to achieve a prescribed terminal accuracy. In terms of the function of the systems, a performance criterion for an aided-inertial guidance system can be formulated as follows.

A guidance and navigation system should operate in such a way that the information from the various sensors is combined and processed in a "most efficient" way, and the guidance policy is chosen in such a manner that the terminal state is reached with a minimum of control effort without violating existing constraints, such as maximum allowable deviations from the nominal position and velocity or constraints imposed on control capability, e.g., maximum throttleability of thrust engine or maximum rate of change of the thrust direction.

This requires the construction of such a mathematical model in the framework of modern system analysis employing optimum estimation and filter theory and guidance schemes based upon the calculus of variation in its classical and/or modern form.

The basic underlying assumptions for such a mathematical model are explained in the following section.

2.1.1 Basic Assumptions for the Mathematical Model

It is the task of this paragraph to develop within the framework developed in the preceding paragraphs, a mathematical model that permits the quantitative specification of a particular guidance and navigation configuration and the quantitative assessment of the system performance. The model should also provide the possibility of synthesizing a system that is optimal with respect to a certain performance index. This system can then be used as a yardstick by which the performance of other systems can be measured.

In the formulation of the mathematical model, two problems must be clearly distinguished. One is the investigation of the performance of guidance systems with respect to statistical ensembles. The other problem is the simulation of actual flights. This distinction is of importance because of the task of determining the validity of the model under real conditions.

In the construction of the mathematical model for the performance assessment of aided-inertial space guidance systems, certain basic assumptions can be made which are valid for all nominal mission profiles and system configurations. These assumptions are first stated in a group.

2.1.1.1 Statement of Basic Assumptions

Assumption 1

Each mission is defined by a nominal mission profile; i.e., position and velocity are prescribed functions of time.

Assumption 2

The deviations from the nominal state as represented by the nominal phase trajectory due to error sources in the guidance, navigation, and control system are small and of first order in all state variables during the time intervals under consideration. The smallness of the deviations from the nominal state during all time intervals that influence the terminal accuracy make it possible to apply first-order perturbation theory.

Assumption 3

Random noise sequences in the plant and observations are assumed to be timewise uncorrelated and Gaussian.

The assumptions have the following implications for the mathematical model as well as on-board mechanization of guidance and navigation schemes.

1. Optimal estimation and prediction theory is applicable, either in the form of maximum liklihood or its theoretical equivalent of Wiener filter theory in its modern version, as developed by Swerling, Kalman and Bucy, Pfeiffer, and Bryson.

2. Optimal guidance schemes are governed by linear differential and difference equations, and the existence of the solution of the corresponding boundary value problem is well established.

3. The "separation principle" become applicable and provides a simple guideline for the synthesis of optimum systems.

4. The performance of the space guidance system concerning terminal accuracy can easily be expressed as function of the covariance matrix P.

2.2 NOMINAL AND ACTUAL TRAJECTORY BLOCKS

2.2.1 Nominal Trajectory

2.2.1.1 Basic Assumptions and General Structure of Trajectory Profile

The model for the nominal re-entry trajectory which provides acceleration, velocity, position, and attitude as a function of time for the performance assessment of aided-inertial re-entry guidance systems, as described below, is based upon the following general assumptions.

a. Physical Environment

   Nonrotating planet with a spherically symmetric gravitational potential and exponential nonrotating atmosphere constitutes the physical environment.

b. Vehicle Configuration

   A rigid lifting vehicle without any thrusting capability outside that required for attitude changes is assumed.

c. Nominal Control

   Aerodynamic control is achieved through a change of the roll angle. The control philosophy is dependent upon the specific phase and is described below in a phenomonological manner.

The mathematical model is formulated in such a fashion that at most seven phases can be encountered in one trajectory. On the other hand, it is possible that the starting point can lie in any of these phases. These phases are schematically depicted in Figure 1.

The model can describe the following major mission profiles:

a. Single-pass trajectories. This class encompasses those trajectories in which the vehicle does not leave the atmosphere after it entered it once.

b. Single-skipout trajectories. This class encompasses those trajectories for which two atmospheric phases are connected with each other by a free-fall orbit.

In addition to these two major classes the model provides the possibility of describing other trajectories such as those encountered in atmospheric braking maneuvers. The latter can be simulated by starting the program in phase 3.

For the sake of clarity, the different phases, as numbered in Figure 1 (i.e., assuming a single-skipout trajectory) are defined as follows.

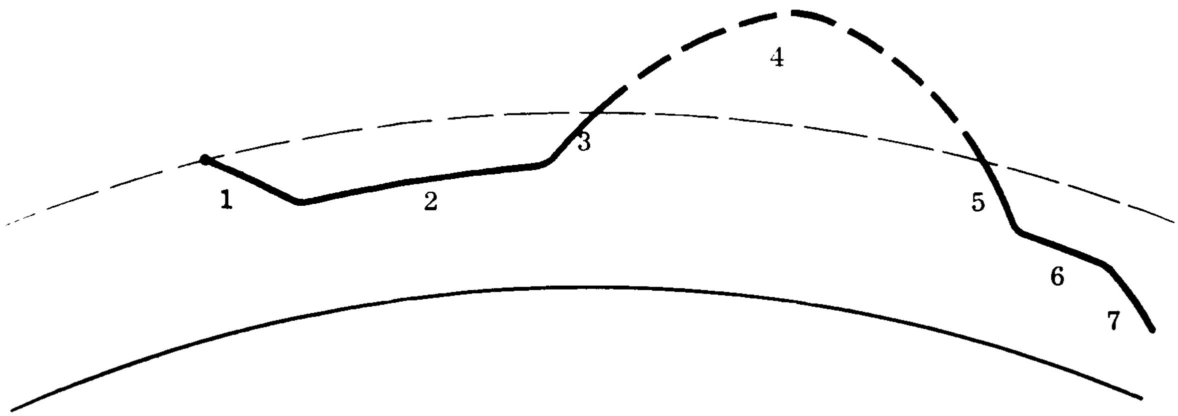


Figure 1. Trajectory Profile (Schematic)

Phase 1: Initial entry phase. A constant roll angle control policy is used during this phase. The sign of the roll angle is allowed to change so that the vehicle will remain sufficiently near to its initial trajectory plane.

Phase 2: "First" constant altitude phase. The roll angle is changed such that the vehicle maintains constant altitude.

Phase 3: Pullout phase. The roll angle is changed according to a prespecified time history. This is done by specifying the coefficients of two second-order polynominals in time. The length of time that the two curve fit control laws are used may be specified. If the vehicle has a high enough speed and appropriate coefficients are specified, a path may be generated such that phase 4 will be entered.

Phase 4: Skiput phase. No trajectory control since it is assumed that thrust is only available for vehicle attitude control.

Phase 5 and 6: Second entry phase and second constant altitude phase. The control policies are equivalent to those in phases 1 and 2, respectively.

Phase 7: Final descent phase. The vehicle is kept at constant roll angle and angle of attack. The roll angle can change signs in order to provide out-of-plane control.

2.2.1.2 Coordinate Systems

The initial position and velocity of the vehicle with respect to the re-entry planet may be input in either cartesian or spherical coordinates. These coordinate systems are shown in Figure 2. The cartesian system is right-handed, irrotational, and orthogonal. The axes may be considered to be oriented with the $\underline{k}$ axis along the northern polar axes of the re-entry planet and the $\underline{i}$ and $\underline{j}$ axis in the equatorial plane. However, this orientation is arbitrary since the planet is assumed to be a nonrotating spherical body. All the velocity and acceleration integrations are performed in the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system.

If the $\underline{i}$, $\underline{j}$, $\underline{k}$ cartesian coordinate system is considered to have the orientation described above, the input spherical coordinate system would have the following meaning:

r_o - radial distance of vehicle from center of re-entry planet

u_o - longitude

λ_o - latitude

V_o - speed

γ_o - flight path angle, measured from the local horizontal plane up to the velocity vector

A_o - azimuth angle, measured clockwise from north to the projection of the velocity vector in the local horizontal plane

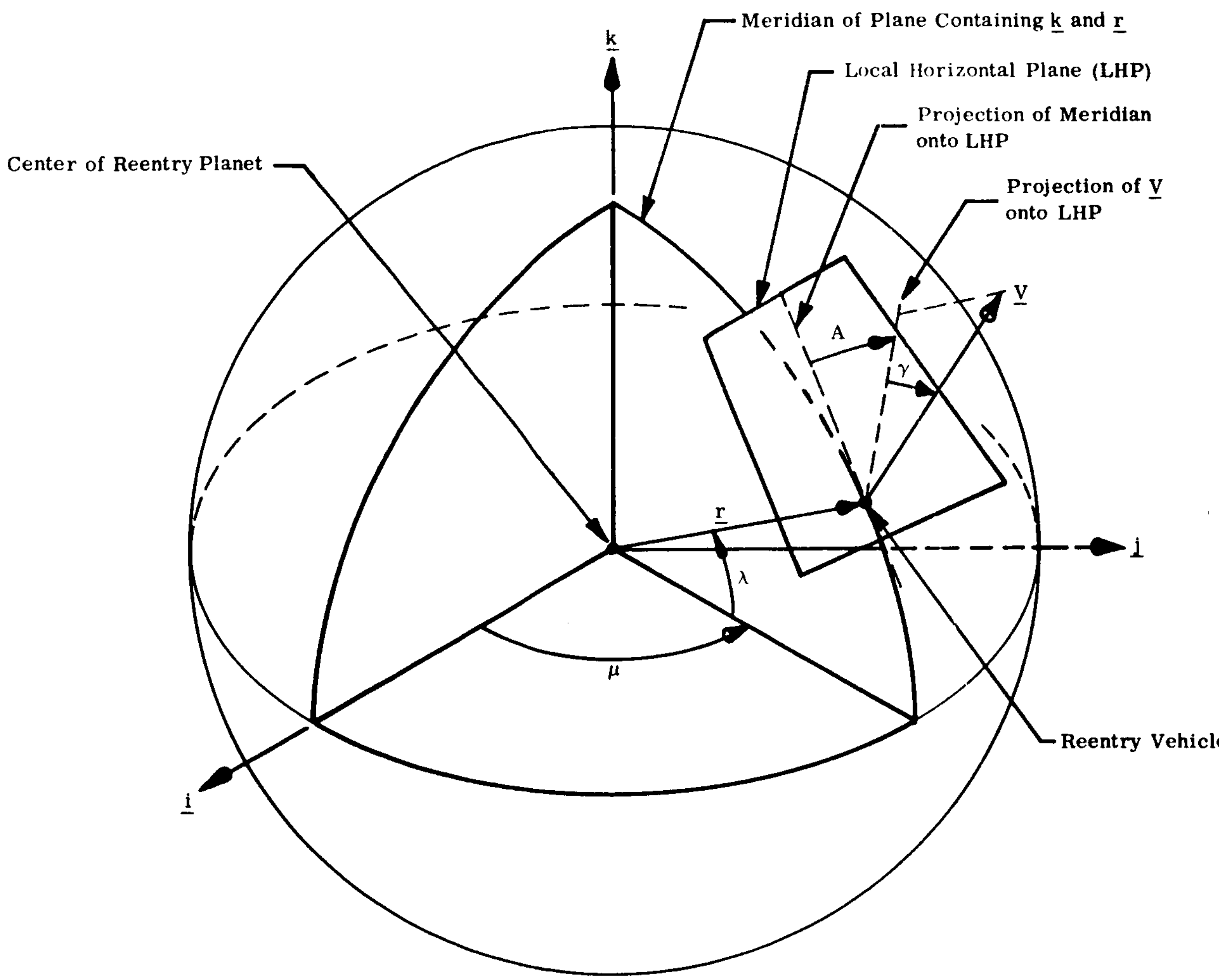
The output position and velocity are given in both the $\underline{i}$, $\underline{j}$, $\underline{k}$ cartesian system and a second spherical coordinate system is shown in Figures 3 and 4. This second spherical system is convenient because it is oriented in the initial trajectory plane. This provides direct observation of out-of-plane velocity and position values.

The output spherical system, r, θ, φ, V, γ, β, is referenced to the right-handed, irrotational, orthogonal cartesian coordinate system, $\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$. The relation between the $\underline{i}$, $\underline{j}$, $\underline{k}$ system and the $\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$ system is shown in Figure 3. The relation between the $\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$ system and the r, θ, φ, V, γ, β system is shown in Figure 4.

The $\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$ system is set up at the beginning of the program ($t = t_o$) with $\underline{j}_t$ and $\underline{k}_t$ forming the initial trajectory plane so that the output spherical system will have the following interpretation:

r - radial distance of vehicle from center of re-entry planet

θ - range angle plus 90 degrees



Input:

TRINP = 1: Position = $X_0\underline{i} + Y_0\underline{j} + Z_0\underline{k}$
Velocity = $\dot{X}_0\underline{i} + \dot{Y}_0\underline{j} + \dot{Z}_0\underline{k}$

TRINP = 0: Position = $r_0,\ \lambda_0,\ \mu_0$
Velocity = $V_0,\ \gamma_0,\ A_0$

Figure 2. Input Coordinate Systems

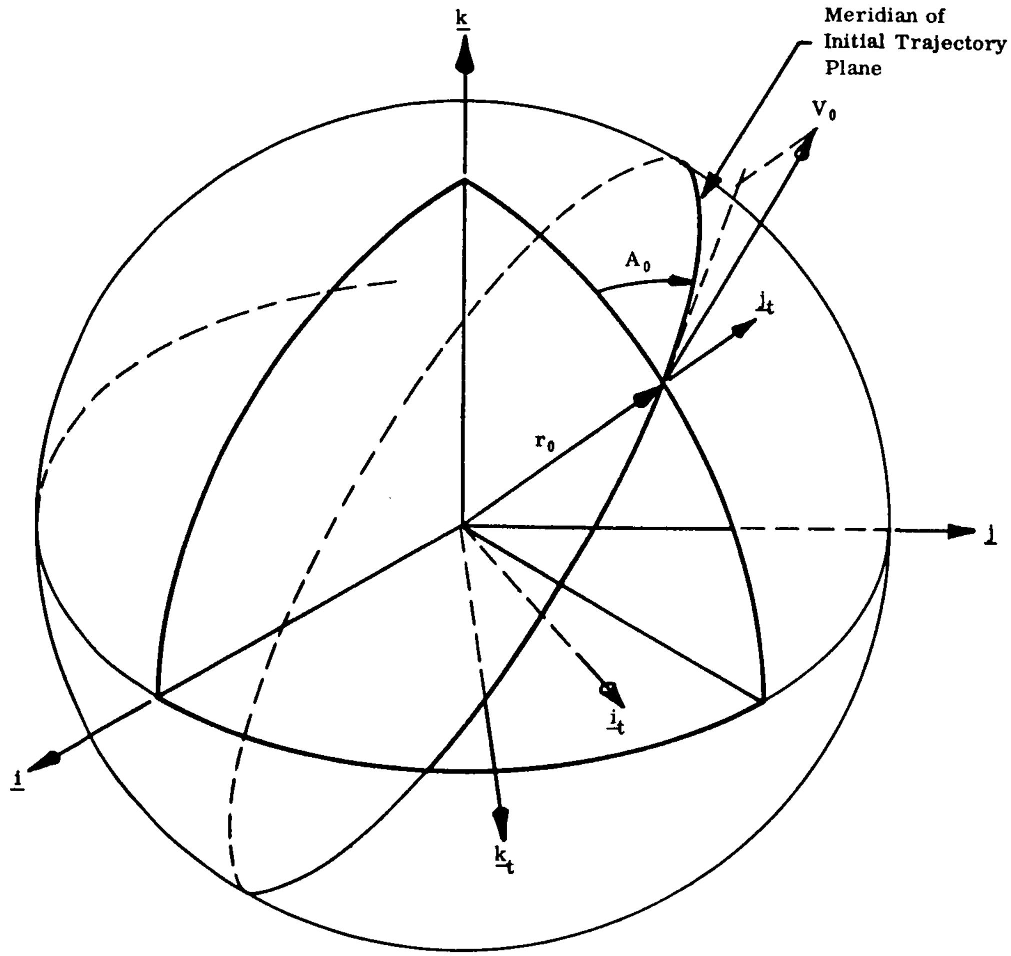
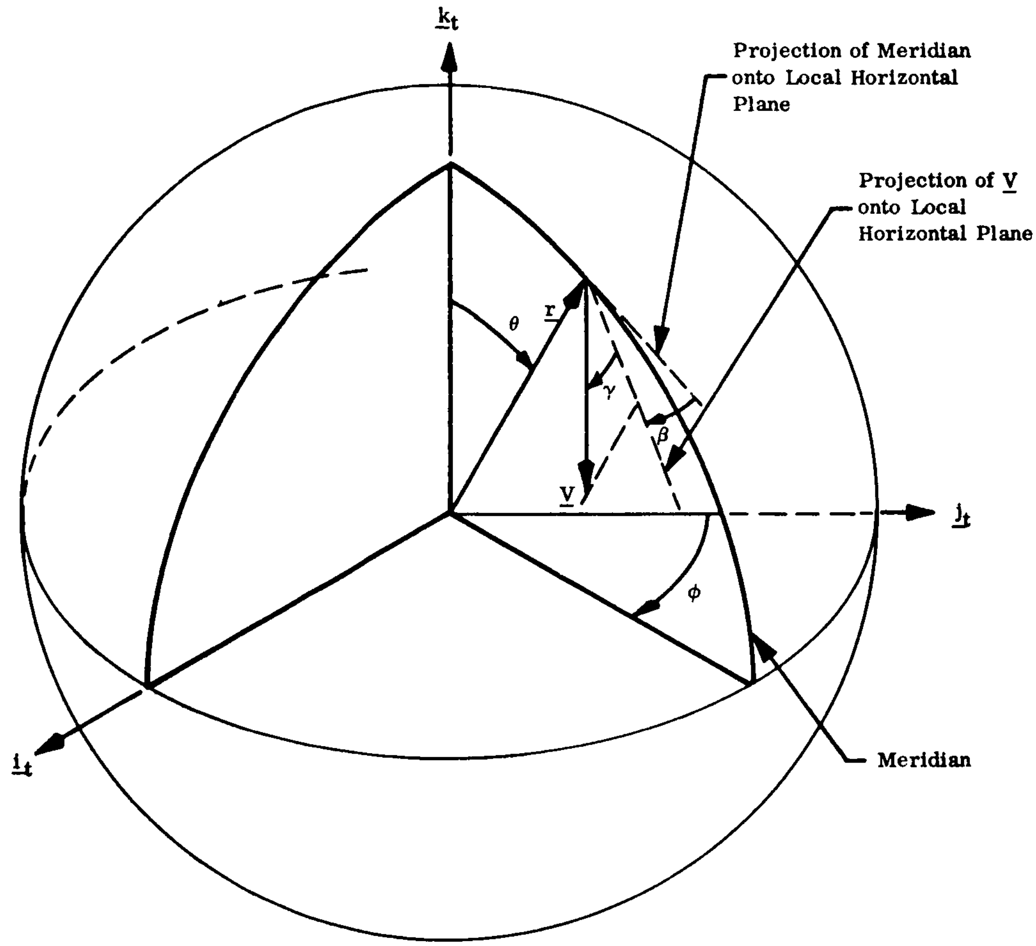


Figure 3. Cartesian Basis for Output Spherical Coordinate System at $t = t_o$ ($\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$) related to $\underline{i}$, $\underline{j}$, $\underline{k}$ system



NOTE: Since θ = range angle +90°, θ is never less than 90°. However, the diagram was drawn as it is for clarity.

The unit vectors $\underline{i}_t$ and $\underline{k}_t$ form the initial trajectory plane.

Figure 4. Relation Between Spherical Output Coordinate System (r, θ, ϕ, V, γ, β) and $\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$ system (at $t \neq t_o$)

φ - out-of-plane position angle

V - speed

γ - flight path angle

β - angle between the projection of the velocity vector onto the local horizontal plane and the plane formed by $\underline{k}_t$ and $\underline{r}$, measured clockwise from the plane

The orientation of the vehicle is given by three orthogonal unit vectors defining the pitch axis ($\underline{P}_I$), the yaw axis ($\underline{Y}_A$), and the roll axis ($\underline{R}_O$). There are three sets of these unit vectors. One set is the actual set ($\underline{P}_I$, $\underline{Y}_A$, $\underline{R}_O$) which describe the current orientation of the vehicle. A second set is the reference set ($\underline{P}_{Io}$, $\underline{Y}_{Ao}$, $\underline{R}_{Oo}$) which is defined with respect to the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates by the matrix transformation (A_4) shown in Block B. 1.4. The third set is the set defining the orientation of the vehicle with respect to the reference set at time = t_o $(\underline{P}_I, \underline{Y}_A, \underline{R}_O)_{t = t_o}$. The initial orientation is different from the reference set by the Euler angles α_{1o}, α_{2o}, α_{3o} (input quantities).

The roll axis ($\underline{R}_O$) is different from the direction of the velocity vector by the angle of attack (α). Also, the pitch axis ($\underline{P}_I$) is rotated from a line perpendicular to the trajectory plane by the angle $90^o - \varphi$. These conditions remain true throughout the flight.

2.2.1.3 Differential Equations of Motion

This section states the differential equations of motion, resulting from the following basic assumptions:

a. The vehicle is represented by a point mass.

   This assumption concerns only the motion of the vehicle along the trajectory. The 6-dimensional dynamical character of the problem is taken into account through the model for the nominal control policy, as described in the next section. It enters the differential equations directly through the control variables which are the roll angle φ, and the angle of attack α.

b. Ablation effects are ignored resulting in the assumption of constant mass.

c. Exponential nonrotating atmosphere and spherical earth are assumed.

Under these assumptions the differential equations of motion in the cartesian ($\underline{i}$, $\underline{j}$, $\underline{k}$) coordinate systems becomes

$$\underline{a} = \frac{1}{M} (\underline{D} + \underline{N}) - \underline{g}$$

$\underline{a}$ is the total acceleration, M the access of the vehicle, $\underline{g} = g_0 \left(\frac{R}{r}\right) \underline{U}_r$ is the gravitational acceleration with

g_o	= sea level gravitational acceleration of re-entry planet
R	= radius of re-entry planet
r	= radial distance of re-entry vehicle from center of re-entry planet
$\underline{U}_r$	= unit position vector of vehicle in the Newtonian reference cartesian coordinate system, $\underline{i}$, $\underline{j}$, $\underline{k}$

The aerodynamic drag forces $\underline{D}$ and the aerodynamic normal force N are given by

$$\underline{D} = - C_D \rho \frac{V^2 S}{2} \underline{U}_v$$

$$\underline{N} = C_N \rho \frac{V^2 S}{2} (\cos \varphi \underline{U}_u - \sin \varphi \underline{U}_p)$$

The dependence of the aerodynamic drag and normal force coefficients C_D and C_N on the angle of attack α is assumed as

$$C_D = C_{D_o} + C_2 \alpha^2 + C_4 \alpha^4$$

$$C_N = C_{N_\alpha} + C_3 \alpha^3 + C_5 \alpha^5$$

where C_{D_o}, C_{N_α}, and C_i $(i = 2, \ldots, 5)$ are properly chosen constants.

The atmosphere density has the form

$$\rho = \rho_o e^{-\beta' (r - R)}$$

V indicates the speed, S the aerodynamic area of the vehicle, and the orthogonal (u, v, p) coordinate system is defined as

$\underline{U}_v$	unit vector in direction of velocity
$\underline{U}_p$	unit vector perpendicular to $\underline{U}_v$ and in instantaneous trajectory, plane
$\underline{U}_u$	$= \underline{U}_u \times \underline{U}_v$

This completes the description of the differential equations in the $(\underline{i}, \underline{j}, \underline{k})$ coordinate system. The equations of motion are always integrated with respect to this coordinate system.

2.2.1.4 Integration Routine

Time is advanced in the program by the integration routine. The integration loop consists of the integration routine (Block I.4 - Range Kutta Type), dynamics block (Block I.2), and the evaluation block (Block I.8). This loop is exited to calculate a new nominal control at each nominal control time, phase change time, print time, and the terminate run time.

2.2.1.5 Nominal Control

Only the roll angle φ is used for aerodynamical control of the vehicle in the different phases as explained in the first section. The quantitative details are discussed phase-wise below.

2.2.1.5.1 Nominal Control During First and Second Re-entry Phase

A constant roll angle is used in these two phases and is changed in sign if the direction of the velocity vector deviates from the initial nominal trajectory plane by more than e_s. Thus

$$\varphi_{ci} = \text{sign}_i \, \varphi_c$$

$$\text{sign}_i = \begin{cases} \text{sign}_{i-1} & \text{if } |\underline{U}_{p_o} \cdot \underline{U}_v| < \epsilon_s \\ \text{sign}\,(\underline{U}_{p_o} \cdot \underline{U}_v) & \text{if } |\underline{U}_{p_o} \cdot \underline{U}_v| \geq \epsilon_s \end{cases}$$

where

$$\underline{U}_{p_o} = (\underline{U}_u \times \underline{U}_v)_{t_o}$$

defines the unit vector normal to the initial trajectory plane.

During a flipover maneuver, the change in the roll angle is assumed to be

$$= \varphi_{i-1} + \omega_{\varphi_{i-1}} \Delta t$$

and the roll rate of the vehicle is computed according to

$$\omega_{\varphi_i} = K_\varphi (\varphi_{ci} - \varphi_i)$$

K_φ is a preselected constant, representing in a gross fashion the vehicle response to the control system.

2.2.1.5.2 Nominal Control During the Constant Altitude Phases

The roll angle control is given by

$$\varphi_c = \text{sign}_i \left[\frac{\pi}{2} + \sin^{-1} (K_1 \Delta\dot{r} + K_2 \Delta r) + \frac{\pi}{2} e^{-K_3 (t - T_c)} \right]$$

where

$\Delta\dot{r} = \dot{r} =$ radial velocity of vehicle

$\Delta r = r - r_c$

$r_c =$ desired constant altitude of vehicle

K_1 and K_2 are gains whose value is either input as constant or calculated as a function of time (optimum gains)

K_3 is an input constant

$T_c =$ time at the beginning of the constant altitude phase

sign_i is chosen in the same fashion as in the initial entry phases and provides out-of-plane control

This control law provides upward normal force $|\varphi_c| < \pi/2)$ as required to keep the vehicle at a constant altitude. The term

$$\frac{\pi}{2} e^{-K_3 (t - T_c)}$$

is used to make $|\varphi_c| = \pi$ at the beginning of the constant altitude. This is helpful in preventing an unintentional skipout.

The following scheme was used to calculate the gains K_1 and K_2 as a function of time: In order that the radial velocity be zero and the radial distance not vary from some desired value (r_c) the following restriction was placed on the command roll angle:

$$\sin(\varphi_c - 90^o) - [K_1 (\dot{r} - 0) + K_2 (r - r_c)] = 0 \qquad (1)$$

Equating the radial acceleration to the acceleration provided by the normal force (normal to the drag force) gives the following:

$$-\Delta\ddot{r} = \frac{1}{M} N \sin(\varphi - 90^o) \qquad (2)$$

where

φ = roll angle of vehicle

M = mass of vehicle

N = magnitude of normal force

$\Delta\ddot{r} = \ddot{r}$ = radial acceleration

Substituting (2) into (1) gives

$$\Delta\ddot{r} + \frac{N}{M} K_1 \Delta\dot{r} + \frac{N}{M} K_2 \Delta r = 0$$

which is analogous to the standard second-order differential equation

$$\ddot{x} + 2\zeta \frac{2\pi}{\tau} \dot{x} + \frac{4\pi^2}{\tau^2} = 0$$

where ζ is the damping ratio and

τ is the natural period of oscillation

Thus we may set

$$K_1 = \frac{4\pi M \zeta}{N\tau} \quad \text{and} \quad K_2 = \frac{4\pi^2 M}{N\tau^2}$$

and input values of ζ and τ such that the constant altitude control policy will have the desired values of damping and oscillation frequency.

2.2.1.5.3 Nominal Control in Pullout Phase

The roll angle is used as a control variable and is specified as

$$\varphi_c = \text{sign}_i [F_0 + F_1 (t - T_c') + F_2 (t - T_c')^2]$$

F_0, F_1, and F_2 are appropriate input quantities; T_C'' the time at beginning of pullout phase. $Sign_i$ is determined as in the initial entry phase and used for out-of-plane control.

2.2.1.5.4 Nominal Control in Final Descent Phase

The roll angle is used as a control variable in the same way as in the initial entry phase.

2.2.1.6 Evaluation

The calculation of heating and pilot acceleration history is performed in the evaluation block (Block I.8). By looking at the output of these values (Q and E_n, respectively) a particular trajectory may be evaluated concerning the severity of ablation on the vehicle and the aerodynamic acceleration effects experienced by the pilot.

Convective heating rate at the stagnation point is calculated as follows.

$$q_c = \frac{C_H}{\sqrt{R_N}} \left(\frac{\rho}{\rho_o}\right)^n \left(\frac{V}{\sqrt{gr}}\right)^m$$

where

q_c	= convective heating rate
C_H	= an input constant whose value depends on the planet's atmosphere and the type of boundary layer flow
n	= an input constant describing boundary flow (n = 0.5 → laminar flow)
m	= an input constant describing the type of flow (m = 3 → laminar flow)
R_N	= radius of curvature of vehicle at stagnation point

Radiative heating rate at the stagnation point is calculated as follows.

$$q_r = k_H R_N \left(\frac{\rho}{\rho_o}\right)^{P_H} C_e V^q$$

where

q_r	= radiative heating rate
k_H	= an input constant specifying the percentage of heat radiation between the gas cap and the vehicle

p_H = an input quantity

$$C_e = \begin{cases} C_{e1} \text{ if } \frac{V}{\sqrt{g_r}} < 1.73 \\ C_{e2} \text{ if } \frac{V}{\sqrt{g_r}} \geq 1.73 \end{cases}$$

C_{e1}, C_{e2} are input quantities

$$q = \begin{cases} q_1 \text{ if } \frac{V}{\sqrt{g_r}} < 1.73 \\ q_2 \text{ if } \frac{V}{\sqrt{g_r}} \geq \end{cases}$$

q_1 and q_2 are input quantities

The total stagnation point heating rate is given by

$$q_s = q_c + q_r$$

The accumulated heat which can be used as a measure of ablative losses is given by

$$Q = \int_{t_o}^{t} q_s \, dt$$

where Q is an output quantity.

The limit of pilot acceleration endurance is represented by

$$\tau' = E_0 + E_1 a' + E_2 (a')^2 + E_3 (a')^3 + E_4 (a')^4$$

$$a' = f/g_e$$

where

τ' = length of time a pilot will remain usefully conscious at a particular acceleration level

a' = aerodynamic acceleration is earth g's

f = aerodynamic acceleration of vehicle

g_e = sea level gravitational acceleration of earth

E_i (i=0,4) = input quantities

The acceleration history of the pilot is represented by

$$E_n = \int_{t_o}^{t} \dot{E}_n dt$$

where

$$\dot{E}_n = \frac{1}{\tau'} \quad \text{if } \frac{1}{\tau'} \geq 0.0008$$

$$\dot{E}_n = 0 \quad \text{if } \frac{1}{\tau'} < 0.0008$$

Thus if the value of E_n ever reaches or exceeds one, the pilot has "lost" useful consciousness. However, the program will not stop on this condition.

2.2.2 Actual Trajectory

2.2.2.1 Basic Assumptions

The model for the actual re-entry trajectory which describes the motion of the vehicle from its perturbed initial state to its terminal state under the influence of the perturbative control vector as generated in the guidance block, is based upon the same set of assumptions used in the nominal trajectory block except for the following modifications or additions.

a. Physical Environment

The atmospheric density, ρ, is written in the nominal trajectory as a function of the sea level density, ρ_o, and the atmospheric decay factor β'.

$$\rho = \rho_o e^{-\beta' h}$$

where h is the altitude of the vehicle.

In the actual trajectory block, the density is written

$$\rho = (\rho_o + \delta\rho_o) e^{-\beta' h}$$

where $\delta\rho_o$ is a random variable assumed to be correlated in altitude.

The correlation of $\delta\rho_o$ is first order and given by the equation

$$\delta \dot{\rho}_o = -\frac{|\dot{h}|}{h_\rho} \delta \rho_o + w_\rho (t)$$

where $w_\rho (t)$ is white noise.

The covariance, $_2Q(t)$, of $w_\rho (t)$ is given by the equation

$$_2Q(t_{p-1}) = |\dot{h}(t_{p-1})| (k_o + [k_1 + k_2 h(t_{p-1})] \; e^{-k_3 [h(t_{p-1}) - h_o]},$$

where the k's and h_o are input constants

Under these conditions the density perturbation, $\delta\rho_o$, is given by the following equation.

$$\delta \rho_o (t_p) = {}_c\Phi (t_p, t_{p-1}) \, \delta \rho_o (t_{p-1}) + {}_c\Gamma_{p,p-1} \, w_\rho (t_{p-1})$$

where

$_c\Phi (t_p, t_{p-1})$ is the solution to the linear homogeneous differential equation for the perturbative density function

and

$_c\Gamma_{p,p-1}$ is the white noise weighting matrix in the density perturbation shaping filter.

b. Vehicle Configuration

The normal and drag force coefficients, C_N and C_D, are modified to read as follows

$$C_N = (C_{N\alpha}{}^* + \delta C_{N\alpha}) \, \alpha + C_3^* \alpha^3 + C_5^* \alpha^5$$

$$C_D = (C_{Do}^* + \delta C_{Do}) + C_2^* \alpha^2 + C_4^* \alpha^4$$

where the * implies values used in the nominal trajectory and the $\delta C_{N\alpha}$ and δC_{Do} are constant random numbers which may be input or computed with a noise generator $\delta C_{N\alpha}$ and δC_{Do} represent the uncertainty in the aerodynamic coefficients of the vehicle.

c. Vehicular Control

Aerodynamic control is accomplished by adding to the roll angle commanded in the nominal trajectory a perturbative roll angle. In addition a perturbative change to the nominal constant angle of attack is made. Both of these control quantities are computed in the Guidance Block (see paragraph 2.5).

2.3 SENSORS

Both electromagnetic and inertial sensors may be used as aiding instruments during planetary atmospheric entry. The electromagnetic sensors consist of ground based instruments, ground trackers, and instruments carried in the vehicle, i.e. horizon sensor and radio altimeter. The inertial sensors comprise an IMU, inertial measurement unit, which is also a vehicle based instrument.

A description of these sensors follows.

2.3.1 Ground Tracking

Three ground trackers may be used. The position of each is defined by means of input ${}_i r_T$, ${}_i\theta$, ${}_i\varphi$ $(i=1,2,3)$ where ${}_i r_T$ is the radial distance to the i^{th} tracker from the center of the planet, ${}_i\theta$ is the longitude of the i^{th} tracker, and ${}_i\rho$ is the latitude. The measurements made by each of these trackers consists of range (ρ), range rate ($\dot{\rho}$), elevation (ψ) and azimuth (η). Figure 5 shows the geometry of the tracker locations and measurements.

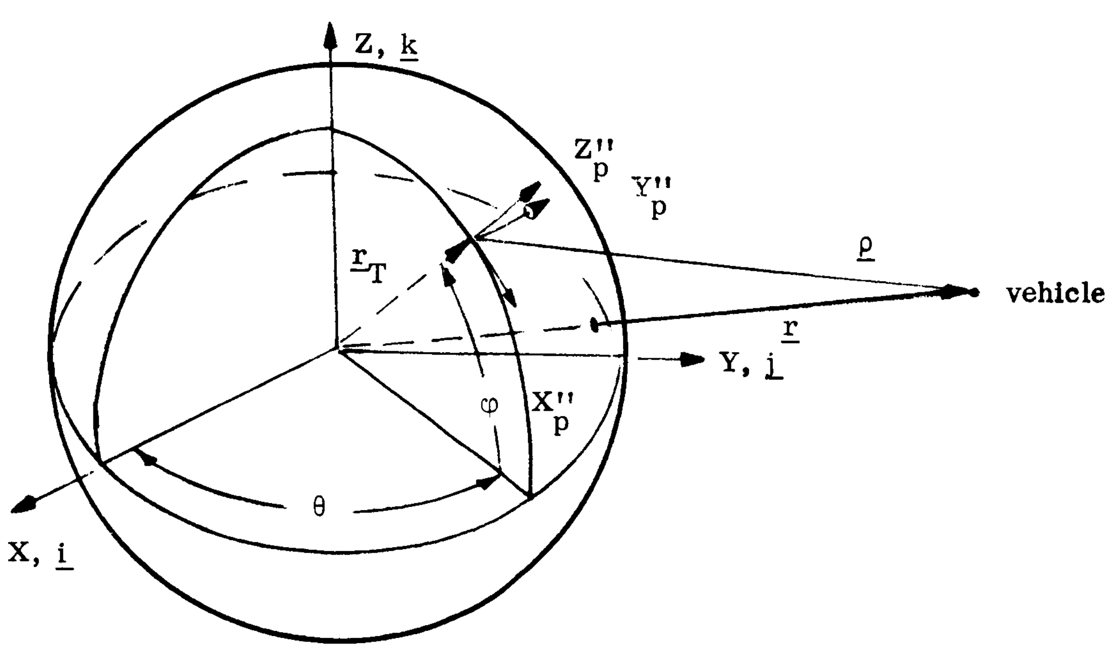


Figure 5. Ground Tracker Coordinate System.

The position of the ground tracker, $\underline{r}_T$, is defined by means of the following equation

$$\underline{r}_T = \cos\varphi \cos\theta\, \underline{i} + \cos\varphi \sin\theta\, \underline{j} + \sin\theta\, \underline{k}$$

The range and range rate vector equations are given by

$$\underline{\rho} = \underline{r} - \underline{r}_T$$

$$\underline{\dot{\rho}} = \underline{\dot{r}} - \underline{\dot{r}}_T = \underline{V}$$

where $\underline{r}$ is the position vector of the vehicle

and $\underline{V}$ is the vehicle's velocity vector

Range and range rate are given by the equations

$$\rho = |\underline{\rho}|$$

$$\dot{\rho} = \frac{\underline{\rho} \cdot \underline{\dot{\rho}}}{|\underline{\rho}|}$$

The elevation of the vehicle with respect to the tracker is given by

$$\psi = \sin^{-1}\left[\frac{\underline{r}_T \cdot \rho}{r_T \rho}\right] \qquad -\frac{\pi}{2} \le \psi \le \frac{\pi}{2}$$

It is convenient to define another coordinate system for use in computing the azimuth angle. This coordinate system is obtained by means of two rotations giving the range vector $\underline{\rho}^{11}$ in the transformed system as a function of the range vector ρ in the X, Y, Z coordinate as shown below

$$\underline{\rho}^{11} = \begin{bmatrix} X''_p \\ Y''_p \\ Z''_p \end{bmatrix} \begin{bmatrix} \sin\varphi & 0 & -\cos\varphi \\ 0 & 1 & 0 \\ \cos\varphi & 0 & \sin\varphi \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \underline{\rho}$$

$$\eta = \tan^{-1}\left[\frac{Y''_p}{-X''_p}\right] \qquad 0 \le \psi < 2\pi$$

The actual and nominal measurements of the i^{th} ground tracker given by

$$ {}_i\underline{Y}^* \triangleq \begin{bmatrix} {}_i\rho^* \\ {}_i\dot{\rho}^* \\ {}_i\psi^* \\ {}_i\eta^* \end{bmatrix} \qquad {}_i\underline{Y} = \begin{bmatrix} {}_i\rho \\ {}_i\dot{\rho} \\ {}_i\psi \\ {}_i\eta \end{bmatrix} $$

where in the actual case $\underline{X}$ replaces the nominal state $\underline{X}^*$ in the computations

Bias errors in the ground tracker measurements have two sources

a. Instrument errors i.e. the instrument makes an error in measuring range, range rate, aximuth or elevation.

b. Tracker location errors i.e. the cartesian coordinates describing the position of the tracker are in error.

These errors may be input or calculated using a noise generator.

2.3.2 Horizon Sensor

A horizon sensor, assumed to be aboard the vehicle, may be used as a navigation instrument. The measurements made by this instrument consist of three angles: elevation (α), azimuth (δ), and half subtended angle (β^H). Figure 6 portrays the geometry associated with this instrument and its measurements.

If the position vector of the vehicle is $\underline{r}$, where

$$ \underline{r} = X_1 \underline{i} + \underline{X}_2 \underline{j} + \underline{X}_3 \underline{k} $$

then the elevation angle, α_1 is given by

$$ \alpha = \sin^{-1} \left[\frac{\underline{X}_3}{|\underline{r}|} \right] \qquad -\frac{\pi}{\alpha} \leq \alpha \leq \frac{\pi}{2} $$

and the azimuth angle, δ, is given by

$$\alpha = -\sin^{-1}\left[\frac{X_3}{|\underline{r}|}\right] \qquad -\frac{\pi}{2} \leq \alpha \leq \frac{\pi}{2}$$

and the azimuth angle, δ, is given by

$$\alpha = \tan^{-1}\left[\frac{X_2}{X_1}\right] \qquad 0 \leq \delta < 2\pi$$

The half subtended angle β^H is written below as a function of the radius of the planet, R_1 and the distance from the vehicle to the planet r.

$$\beta^H = \sin^{-1} \left[\frac{R}{r}\right]$$

Notice that the angular measurements originate at the vehicle. See Figure 6.

The actual and nominal measurements of the horizon sensor are given by

$$_4\underline{Y}^* \triangleq \begin{bmatrix} \alpha^* \\ \delta^* \\ \beta^H \end{bmatrix} \qquad _4Y \triangleq \begin{bmatrix} \alpha \\ \delta \\ \beta^H \end{bmatrix}$$

where in the actual case $\underline{X}$ replaces the nominal state X^* in the computations.

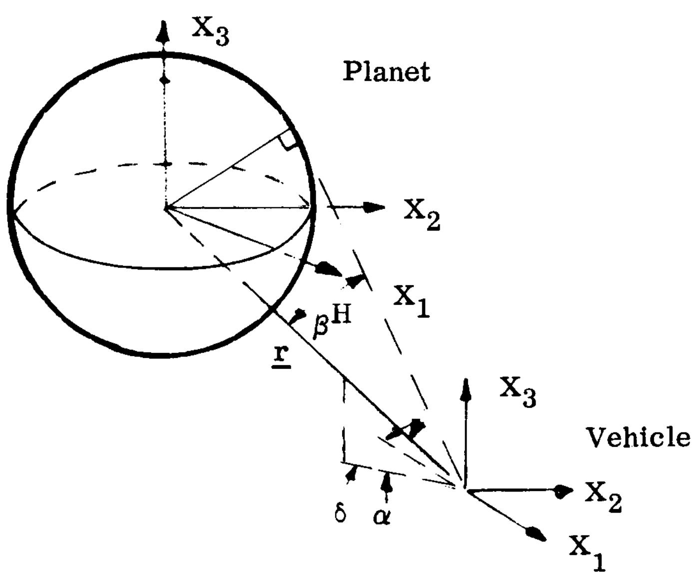


Figure 6. Horizon Sensor Coordinate System

Bias errors for this instrument consist of errors in the three angular measurements made by the instrument. These may be input or calculated with a noise generator.

2.3.3 Radio Altimeter

The last electromagnetic sensor, the radio altimeter, makes on board measurements of altitude, h, and radial speed, $\dot{r}$.

The altitude is the difference between the radial distance to the vehicle, r, and the planet's radius, R, since the planet is spherical.

$$h = r - R$$

Radial speed, $\dot{r}$, is the components of the vehicle's velocity along the radial direction.

$$\dot{r} = \frac{\underline{r} \cdot \underline{V}}{r}$$

The nominal and actual measurements of the radio altimeter are given by

$$_6\underline{Y}^* = \begin{bmatrix} h^* \\ \dot{r}^* \end{bmatrix} \qquad _6\underline{Y} = \begin{bmatrix} h \\ \dot{r} \end{bmatrix}$$

where in the actual case $\underline{X}$ replaces the nominal state $\underline{X}^*$ in the computations.

Bias errors for this instrument consist or errors in the measurement of h and r and are input or calculated with a noise generator.

2.3.4 Inertial Measurement Unit

The inertial measurement unit senses aerodynamic acceleration. It consists of three gyros to supply attitude information and three accelerometers. On an optional basis the IMU can perform as a gimballed system i.e. one in which the stabilization gyros and accelerometers are isolated from the angular motion of the vehicle by means of three gimbals or it may function as a strap down system i.e. one which has the accelerometers and gyros tied to the vehicle frame.

The measurements made by the IMU consist of components of integrated aerodynamic acceleration. These are computed in block I for the nominal trajectory and block IV for the actual. They are identified respectively as

$$_7Y^* = \int_{t_o}^{t} \underline{f}^* = \begin{bmatrix} \int a_x^* \\ \int a_y^* \\ \int a_z^* \end{bmatrix} \qquad _7Y = \int_{t_o}^{t} \underline{f} = \begin{bmatrix} \int a_x \\ \int a_y \\ \int a_z \end{bmatrix}$$

In contrast to the electromagnetic sensors, the IMU bias errors are always calculated. The error model for this is described in the following paragraphs.

Because the bias errors are included in the augmented state and because the computational load and computer storage requirements rise very rapidly as a function of the dimension of the state, the error model was kept very simple. There are 15 bias errors distributed among three gyros and three accelerometers according to the following schedule.

$\epsilon_1, \epsilon_4, \epsilon_1$	Initial misalignment of gyros 1, 2, 3
$\epsilon_2, \epsilon_5, \epsilon_8$	Constant drift of gyros 1, 2, 3
$\epsilon_3, \epsilon_6, \epsilon_9$	Acceleration dependent drift of gyros 1, 2, 3
$\epsilon_{10}, \epsilon_{12}, \epsilon_{14}$	Bias errors of accelometers 1, 2, 3
$\epsilon_{11}, \epsilon_{13}, \epsilon_{15}$	Scale factor errors of accelometers 1, 2, 3

The gyros and accelerometers are oriented with respect to each other in an invarient configuration defined in table below. The X_1, Y_1, Z_1 define an orthogonal right handed triad.

	X_1	Y_1	Z_1
gyro 1	input	output	spin
gyro 2	spin	input	output
gyro 3	output	spin	input
accel 1	sensitive		
accel 2		sensitive	
accel 3			sensitive

Table 1. Gyro and Accelerometer Orientation

In the gimballed system the X_1, Y_1, Z_1 coordinate system is related to the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system by means of the following transformation

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = [M]\ [C_o] \begin{bmatrix} \underline{i} \\ \underline{j} \\ \underline{k} \end{bmatrix}$$

where

C_o is a transformation relating the reference body axes P_{Io}, Y_{Ao}, R_{oo} to the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system

and

M is an input transformation relating the instrument axes to the <u>reference</u> body axes

When the strap down configuration is called for, the X_1, Y_1, Z_1 coordinate system by means of the following transformation

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = [M]\ [C] \begin{bmatrix} \underline{i} \\ \underline{j} \\ \underline{k} \end{bmatrix}$$

where

M is an input transformation relating the instrument axes to the <u>current</u> body axes

and

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_3 & \sin\alpha_3 \\ 0 & \sin\alpha_3 & \cos\alpha_3 \end{bmatrix} \begin{bmatrix} \cos\alpha_2 & 0 & -\sin\alpha_2 \\ 0 & 1 & 0 \\ \sin\alpha_2 & 0 & \cos\alpha_2 \end{bmatrix} \begin{bmatrix} \cos\alpha_1 & \sin\alpha_1 & 0 \\ -\sin\alpha_1 & \cos\alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} C_o$$

where

α_1, α_2, α_3 are the inner, middle and outer gimbals angles respectively

and

C_o is as defined above.

The drift of the i^{th} gyro about its input axis, ϕ_i, is written below as a function of the K's which are normalizing constants, ϵ's which are bias errors of the gyros defined earlier and a_i $(i = 1, 2, 3)$ which are components of aerodynamic acceleration along the input axes of the gyros and may be computed from $\underline{f}$ by the following equation.

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = [M] \; [C] \; \underline{f}$$

The acceleration errors caused by gyro drift are computed in the IMU error matrices section, integrated and stored on tape since they are a function only of the nominal trajectory. These acceleration errors, $\Delta \underline{a}_g$, are a function of the aerodynamic acceleration, $\underline{f}$, and the drift of the gyros $\underline{\phi}$, the latter assumed to be small. Under these conditions

$$\Delta \underline{a}_g = \underline{\phi} \times \underline{f}$$

where

$$\underline{\phi} = \phi_1 \underline{X}_1 + \phi_2 \underline{X}_2 + \phi_3 \underline{X}_3$$

and

$$\phi_1 = \epsilon_1 K_1 + \epsilon_2 K_2 \int_{t_o}^{t} d\tau + \epsilon_3 K_3 \int_{o}^{t} a_1 d\tau$$

$$\phi_2 = \epsilon_4 K_1 + \epsilon_5 K_2 \int_{t_o}^{t} d\tau + \epsilon_6 K_3 \int_{t_o}^{t} a_2 d\tau$$

$$\phi_3 = \epsilon_7 K_1 + \epsilon_8 K_2 \int_{t_o}^{t} d\tau + \epsilon_9 K_3 \int_{t_o}^{t} a_3 d\tau$$

and

the $\underline{X}_i$ are expressed in terms of $\underline{i}$, $\underline{j}$, $\underline{k}$

The acceleration errors due to bias and scale factor accelerometers are resolved along the input or sensitive axes of the accelerometers. These errors, $\Delta \underline{a}'_a$ may be written as follows.

$$\Delta \underline{a}'_a = \begin{bmatrix} \epsilon_{10} K_4 + \epsilon_{11} K_5 a_1 \\ \epsilon_{12} K_4 + \epsilon_{13} K_5 a_2 \\ \epsilon_{14} K_4 + \epsilon_{15} K_5 a_3 \end{bmatrix}$$

This vector is resolved along the $\underline{i}$, $\underline{j}$, $\underline{k}$ axes giving $\Delta \underline{a}$

$$\Delta \underline{a}_a = [C]^T [M]^T \Delta \underline{a}'$$

where the superscript T specifies transpose.

The total error in the acceleration measurement is a combination of the errors due to the gyros and the accelerometers.

The instrument uncertainties, ϵ_i, are factored and the uncertainty in the acceleration measurement, $\Delta \underline{a}$, is

$$\Delta \underline{a} = \Delta \underline{a}_g + \Delta \underline{a}_a$$

$$\Delta \underline{a} = [G_{11}\ G_{21}\ G_{31}\ G_{12}\ G_{22}\ G_{32}] \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_{15} \end{bmatrix}$$

where

G_{i1} are (3 x 3) matrices giving errors in acceleration due tothe i^{th} gyro for a unit uncertainty in the ϵ's

and

G_{i2} are (3 x 2) matrices giving errors in acceleration due to the i^{th} accelerometer for a unit uncertainty in the ϵ's

The error in the measurement then, which is integrated non-gravitational acceleration can be written as:

$$\int_{t_o}^{t} \Delta \underline{a} = \left[\int_{t_o}^{t} G_{11} dt \ \int_{t_o}^{t} G_{21} dt \ \int_{t_o}^{t} G_{31} dt \ \int_{t_o}^{t} G_{12} dt \ \int_{t_o}^{t} G_{22} dt \ \int_{t_o}^{t} G_{32} dt \right] \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_{15} \end{bmatrix}$$

2.4 NAVIGATION

It is the function of the navigation system to establish the state of the vehicle where a part of this state consists of position and velocity. This is accomplished by combining measurements made by the sensors in an optimal fashion using a Kalman Filter. Redundant measurements are required since the navigation scheme is based on linear theory and because the measurements are subject to both noise and bias errors.

The output of this block is a best estimate of the state, $\Delta\hat{\underline{x}}_k$, which represents the best estimate of the deviation of the actual trajectory from the nominal. Guidance is generated using this best estimate of the position and velocity.

The following paragraphs outline the development of the Kalman Filter equations.

2.4.1 Non-linear Equations of Motion

The equations of motion of a point mass entering the atmosphere of a spherically symetric non-rotating planet are presented in this section. The equations can be written in general form and have the following form.

$$\ddot{r} = g(\underline{r}) + f\,(\underline{r}, \dot{\underline{r}}, \underline{U}, \underline{A}^F, A^V) \qquad 2.4.1$$

where $\ddot{\underline{r}}$, $\dot{\underline{r}}$, and $\underline{r}$ represent the acceleration, velocity, and position respectively of the spacecraft relative to some inertially fixed, cartesian coordinate system. The term $\underline{g}(\underline{r})$ contains the gravitational acceleration. Atmospheric effects are described by $\underline{f}(\underline{r}, \dot{\underline{r}}, \underline{U}, \underline{A}^F)$. The control term $\underline{U}$ is p-dimensional. The vector $\underline{A}^F$ and $\underline{A}^V$ have been included in order to define variables that appear in a system and whose values are not known precisely. A^F is a two dimensional vector whose components represent constant uncertainties in the aerodynamic coefficients (C_{Do} and $C_{N\alpha}$) of the vehicle. A^V describes the uncertainty in the planetary atmosphere ($\delta\rho_o$).

The first item that should be observed about 2.4.1 is that it is a second-order vector differential equation. It is reduced to a system of first-order equations by introducing the following definitions.

$$\begin{aligned} \underline{X}^p &\triangleq \underline{r} \\ \underline{X}^v &\triangleq \dot{\underline{r}} \end{aligned} \qquad 2.4.2$$

Equation 2.4.1 can now be written as a system of six, first-order equations

$$\underline{\dot{X}} = \begin{bmatrix} \underline{\dot{X}}^p \\ \underline{\dot{X}}^v \end{bmatrix} = \begin{bmatrix} \underline{X}^v \\ g(\underline{X}^p) + \underline{f}(\underline{X}^p, \underline{X}^v, \underline{U}, \underline{A}^F, \underline{A}^V) \end{bmatrix} \qquad 2.4.3$$

In spherical coordinates equation 2.4.3 has the following form

$$\underline{\dot{X}}^s = \begin{bmatrix} \underline{\dot{X}}^{ps} \\ \underline{\dot{X}}^{vs} \end{bmatrix} \begin{bmatrix} \underline{h}^s(\underline{X}^{ps}, \underline{X}^{vs}) \\ \underline{g}^s(\underline{X}^{ps}) + \underline{f}^s(\underline{X}^{ps}, \underline{X}^{vs}, U, \underline{A}^F, A^V) \end{bmatrix} \qquad 2.4.4$$

where

the addition of the s superscript implies spherical coordinates

and

the vector function $\underline{h}^s$ relates components of position and velocity to the time derivatives of the components of position.

Using the spherical coordinate system described in Figure 4, the components of position and velocity are respectively r, θ, φ, V, γ, β. Equation 2.4.4 is presented below as a function of these components.

$$\begin{bmatrix} \dot{r} \\ \dot{\theta} \\ \dot{\phi} \\ \dot{V} \\ \dot{\gamma} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} V \sin\gamma \\ (V/r)\cos\gamma\cos\beta \\ \dfrac{V\cos\gamma\sin\beta}{r\sin\theta} \\ \dfrac{D}{M} - g\sin\gamma \\ \dfrac{N\cos\varphi}{MV} + \left(\dfrac{V}{r} - \dfrac{g}{V}\right)\cos\gamma \\ \dfrac{N\sin\varphi}{MV} + \dfrac{V\cos\gamma\sin\beta\cos\theta}{r\sin\theta} \end{bmatrix} \qquad 2.4.5$$

where

$g = go\left(\dfrac{R}{r}\right)^2$ gravitational force

$D = C_D\,\rho\,\dfrac{V^2 S}{2}$ aerodynamic drag

$N = C_N\,\rho\,\dfrac{V^2 S}{2}$ aerodynamic lift

$$\rho = (\rho_o + \delta\rho_o)\, e^{-\beta'(r - R)} \quad \text{atmospheric density}$$

$$C_D = (C_{D_o} + \delta C_{D_o}) + C_2\alpha^2 + C_4\alpha^4 \quad \text{drag coefficient}$$

$$C_N = (C_{N\alpha} + \delta C_{N\alpha})\,\alpha + C_3\alpha^3 + C_5\alpha^5 \quad \text{lift coefficient}$$

and

M is the vehicle mass, α the angle of attack, φ the bank angle. R, g_o, ρ_o, β' are input constants. δC_{D_o} and $\delta C_{N\alpha}$ are the uncertainties in C_{D_o} and $C_{N\alpha}$ which in turn are the components of $\underline{A}^F$. $\delta\rho_o$ is the uncertainty in sea level atmospheric density, ρ_o, which is the component of $\underline{A}^V$.

2.4.2 Linear Perturbation Equation

Using techniques described in Reference 15, a linear perturbation equation is obtained from the systems of 6 first order nonlinear equations shown in Equations 2.4.3 or 2.4.4. The linear perturbation equation is

$$\underline{x} = F_1(t)\,\underline{x} + F_2(t)\,\underline{x} + E_2(t)\,\underline{b}' + E_3(t)\,\underline{c} + E_4(t)\,\underline{u} \qquad 2.4.6$$

or

$$\underline{x}^s = F_1^s(t)\,\underline{x}^s + F_2^s(t)\,\underline{x}^s + E_2^s(t)\,\underline{b}' + E_3^s(t)\,\underline{c} + E_4^s(t)\,\underline{u} \qquad 2.4.7$$

for the cartesian or spherical systems respectively.

Appropriate terms of the spherical set are defined below.

$$F_1^s(t) \triangleq \begin{bmatrix} \dfrac{\partial \underline{h}^s}{\partial \underline{X}^{ps}} & \dfrac{\partial \underline{h}^s}{\partial \underline{X}^{vs}} \\[2ex] \dfrac{\partial \underline{g}^s}{\partial \underline{X}^{ps}} & \dfrac{\partial \underline{g}^s}{\partial \underline{X}^{vs}} \end{bmatrix} \qquad 2.4.8$$

$$F_2^s(t) \triangleq \begin{bmatrix} 0 & 0 \\[1ex] \dfrac{\partial \underline{f}^s}{\partial \underline{X}^{ps}} & \dfrac{\partial \underline{f}^s}{\partial \underline{X}^{vs}} \end{bmatrix} \qquad 2.4.9$$

$$E_2^s(t) \triangleq \begin{bmatrix} 0 \\ \dfrac{\partial f}{\partial \underline{A}^F} \end{bmatrix} \qquad 2.4.10$$

$$E_3^s(t) \triangleq \begin{bmatrix} 0 \\ \dfrac{\partial \underline{f}^s}{\partial \underline{A}^V} \end{bmatrix} \qquad 2.4.11$$

$$E_4^s(t) \triangleq \begin{bmatrix} 0 \\ \dfrac{\partial \underline{f}^s}{\partial \underline{U}} \end{bmatrix} \qquad 2.4.12$$

$\underline{x}^s \triangleq \underline{X}^s - \underline{X}^{*s}$ where * denotes nominal trajectory value

$$\underline{b}' \triangleq \begin{bmatrix} \delta C_{Do} \\ \delta C_{N\alpha} \end{bmatrix} \qquad 2.4.13$$

$$\underline{c} \triangleq \delta \underline{\rho}_o \qquad 2.4.14$$

$$\underline{u} \triangleq \begin{bmatrix} \delta\alpha \\ \delta\varphi \end{bmatrix} \qquad 2.4.15$$

2.4.3 Linear Difference Equation

The solution to the linear perturbation equation can be expressed as a linear difference equation relating values of the state at $t = t_k$ to values at $t = t_{k-1}$. In the cartesian coordinate system, this equation looks like

$$\underline{x}_k = \Phi_{k,k-1}\, \underline{x}_{k-1} + B_{k,k-1}\, \underline{b}' + C_{k,k-1}\, \underline{c}_{k-1} + \Gamma_{k,k-1}\, \underline{u}_{k-1} \qquad 2.4.16$$

in the spherical system the superscript s is appended giving

$$\underline{x}_k^s = \Phi^s_{k,k-1}\, \underline{x}^s_{k-1} + B_{k,k-1}\, \underline{b}' + C^s_{k,k-1}\, \underline{c}_{k-1} + \Gamma^s_{k,k-1}\, \underline{u}_{k-1} \qquad 2.4.17$$

where

$$\Phi^{s}_{k,k-1} \triangleq \int_{t_{k-1}}^{t_k} \dot{\Phi}^{s}(t,t_{k-1})\, dt = \int_{t_{k-1}}^{t_k} \left[F_1^{s}(t) + F_2^{s}(t)\right] \Phi^{s}(t,t_{k-1})\, dt$$

$$_c\Phi_{k,k-1} \triangleq \int_{t_{k-1}}^{t_k} {}_c\dot{\Phi}(t,t_{k-1})\, dt = \int_{t_{k-1}}^{t_k} -\frac{|\dot{r}|}{h_p}\, {}_c\Phi(t,t_{k-1})\, dt$$

$$B^{s}_{k,k-1} \triangleq \int_{t_{k-1}}^{t_k} \Phi^{s}(t_k,t)\, E_2^{s}(t)\, dt$$

$$C^{s}_{k,k-1} \triangleq \int_{t_{k-1}}^{t_k} \Phi^{s}(t_k,t)\, E_3(t)\, {}_c\Phi(t,t_{k-1})\, dt$$

$$\Gamma^{s}_{k,k-1} \triangleq \int_{t_{k-1}}^{t_k} \Phi(t_k,t)\, E_4(t)\, dt$$

Matrices $\Phi^{s}_{k,k-1}$, $B^{s}_{k,k-1}$, $C^{s}_{k,k-1}$, $\Gamma^{s}_{k,k-1}$ are evaluated along the nominal trajectory. Because the navigation equations are to be calculated in the cartesian coordinate system, matrices satisfying equation 12 must be evaluated. This is accomplished by using the matrix transformation A_6 defined in paragraph and the $\Phi^{s}_{k,k-1}$, $B^{s}_{k,k-1}$, $C^{s}_{k,k-1}$, $\Gamma^{s}_{k,k-1}$ just described. This is accomplished as shown below.

Premultiply equation 2.4.17 by $A_6^{-1}(t_k)$

Replace $\underline{x}^{s}_{k-1}$ by $A_6(t_{k-1})\, \underline{x}_{k-1}$

$$\underline{x}_k = \left[A_6^{-1}(t_k)\, \underline{x}^{s}_k\right] = \left[A_6^{-1}(t_k)\, \Phi^{s}_{k,k-1}\, A_6(t_{k-1})\right] \underline{x}_{k-1} + \left[A_6^{-1}(t_k)\, B^{s}_{k,k-1}\right] \underline{b}'$$

$$+ \left[A_6^{-1}(t_k)\, C_{k,k-1}\right] \underline{c}_{k-1} + \left[A_6^{-1}(t_k)\, \Gamma^{s}_{k,k-1}\right] \underline{u}_{k-1} \qquad 2.4.18$$

Equation 2.4.18 is identical to equation 2.4.16 if the following substitutions are made.

$$\Phi_{k,k-1} = A_6^{-1}(t_k)\,\Phi^s_{k,k-1}\,A_6(t_{k-1})$$

$$B_{k,k-1} = A_6^{-1}(t_k)\,B^s_{k,k-1}$$

$$C_{k,k-1} = A_6^{-1}(t_k)\,C^s_{k,k-1}$$

$$\Gamma_{k,k-1} = A_6^{-1}(t_k)\,\Gamma_{k,k-1}$$

2.4.4 Treatment of Measurements

Navigation is accomplished by comparing the difference between measurements made with electromagnetic sensors on the actual trajectory, $\underline{Y}(t)$, and the nominal trajectory, $\underline{Y}^*(t)$, with the difference in these measurements predicted on the basis of linear theory. The comparison is degraded by the addition of noise and instrument bias errors to the measurements. This results in the measurement having the form

$$\underline{z}_k = \underline{Y}_k - \underline{Y}^*_k + \underline{b} + \underline{v}_k \qquad 2.4.19$$

where

$\underline{b}$ = constant instrument bias errors

$\underline{v}_k$ = white noise sequence

when the bias error is in the measurement. In some cases the bias error is in the location or function of the instrument requiring the use of a transformation, H_B relating the bias errors to measurement errors. In this case the measurement has the form

$$\underline{z}_k = \underline{Y}_k - \underline{Y}^*_k + H_B\,\underline{b} + \underline{v}_k \qquad 2.4.20$$

In the re-entry program, state-related measurements are made with the electromagnetic sensors. The linear model of these measurements has the form

$$\underline{y}_k = H_k\,\underline{x}_k + H_b\,\underline{b} \qquad 2.4.21$$

where

$$H_k \triangleq \left[\frac{\partial \underline{h}}{\partial \underline{X}}\right] \qquad 2.4.22$$

if

$$\underline{Y}(t) = \underline{h}(\underline{X}) \qquad 2.4.23$$

The IMU makes acceleration - related measurements. The linear model of this measurement, $\underline{S}_k$, is outlined below.

$$\underline{S}_k = {}_aJ_k \, \underline{x}_k + {}_2J_k \, \underline{b}' + {}_3J_k \, \underline{C}_k + \underline{\eta}_k + \underline{\sigma}_k$$

where

$${}_aJ_k = {}_aJ_{k-1} \, \Phi(t_{k-1}, t_k) \int_{t_{k-1}}^{t_k} F_2(2) \, \Phi(t, t_k) \, dt$$

$${}_2J_k = {}_2J_{k-1} - {}_aJ_{k-1} \, \Phi(t_{k-1}, t_k) \, \beta_{k,k-1}$$

$$+ \int_{t_{k-1}}^{t_k} F_2(t) \int_{t_k}^{t} \Phi(t, \tau) \, E_2(\tau) \, d\tau \, dt + \int_{t_{k-1}}^{t_k} E_2(t) \, dt$$

$${}_3J_k = {}_3J_{k-1} - {}_aJ_{k-1} \, \Phi(t_{k-1}) \, C_{k,k-1} \, {}_c\Phi(t_{k-1}, t_k)$$

$$+ \int_{t_{k-1}}^{t_k} F_2(t) \int_{t_k}^{t} \Phi(t, \tau) \, E_3(\tau) \, {}_c\Phi(\tau, t_k) \, d\tau \, dt \, +$$

$$+ \int_{t_{k-1}}^{t_k} E_3(t) \, {}_c\Phi(t, t_k) \, dt$$

$$\gamma_k = - {}_aJ_{k-1} \, \Phi(t_{k-1}, t_k) \, \Gamma(t_k, t_{k-1}) + \int_{t_{k-1}}^{t_k} E_4(t) \, dt$$

$$+ \int_{t_{k-1}}^{t_k} F_2(t) \int_{t_k}^{t} \Phi(t, \tau) \, G(\tau) \, d\tau \, dt$$

$$\underline{\sigma}_k = \underline{\sigma}_{k-1} + \gamma_k \, \underline{u}_{k-1}$$

Since the J matrices are functions of F_1, F_2, Φ etc which are more easily expressed in spherical coordinates, the J's are evaluated first in spherical coordinates and transformed to cartesian in an analogous fashion to the β, C, Γ etc of the previous section.

2.4.5 Optimal Linear Estimation (Kalman Filter)

The measurements are contaminated by noise. The adverse effect of this noise can be mitigated if redundant measurements are made and combined in an optimal fashion. This is done in the Kalman filter.

The estimate of the state at t_k is computed as a linear combination of the estimate $\underline{x}(t_{k-1})$ and the measurements $\underline{z}(t_k)$.

$$\underline{x}(t_k) = \Phi(t_k, t_{k-1}) \, \underline{x}(t_{k-1}) + K(t_k) \, [\underline{z}(t_k) - H(t_k) \, \Phi(t_k, t_{k-1}) \, \underline{x}(t_k)]$$

The gain matrix $K(t_k)$ is chosen to minimize the expected value of the sum of the squares of the error in the estimate.

$$E \left\{ [\underline{x}(t_k) - \underline{x}(t_k)]^T \, [\underline{x}(t_k) - \underline{x}(t_k)] \right\} = \sum_{i=1}^{n} (\underline{x}_i(t_k) - x_i(t_k))^2$$

It is found to be

$$K(t_k) = P'(t_k) \, H^T(t_k) \, [H(t_k) \, P'(t_k) \, H^T(t_k) + R(t_k)]^{-1}$$

$$P'(t_k) = \Phi(t_k, t_{k-1}) \, P(t_{k-1}) \, \Phi^T(t_k, t_{k-1})$$

$$P(t_k) = [I - K(t_k) \, H(t_k)] \, P'(t_k) \, [I - K(t_k) \, H(t_k)]^T + K(t_k) \, R(t_k) \, K^T(t_k)$$

$R(t_k)$ is the covariance of the noise in the measurements at t_k.

2.5 GUIDANCE

It is the function of the guidance system to determine the control action that is required to satisfy prespecified trajectory constraints. In this instance, the angle of attack and roll angle must be chosen so that the deceleration of the spacecraft does not exceed certain limits during the flight and must also cause terminal constraints on the position and velocity to be satisfied.

The guidance problem is posed in terms of determining the perturbed control, $\underline{u}_c$, at time $t = t_c$. The actual trajectory contains certain stochastic effects. For this reason, it is more practical to minimize the deviation in some sense and only require that the vehicle be in the neighborhood of the target. This aspect is further emphasized because terminal control laws generally tend to become unstable as the terminal time is approached and some rather violent maneuvers could be commanded.

The control policy used in the re-entry trajectory program is called Lambda Matrix Control. It operates by choosing the perturbed control, $\underline{u}_c$, as a function of the measurement data so that the expected value of the performance index

$$V_N = \sum_{i=1}^{N} (\underline{x}_i^T W_i^X \underline{x}_i + \underline{u}_{i-1}^T W_i^U \underline{u}_{i-1}) \qquad 2.5.1$$

is minimized. The non-negative definite, symmetric matrices W_i^X and W_{i-1}^U are arbitrary and can be selected to limit the amount of control and/or state perturbation along the trajectory.

The control perturbation is determined from the estimate $\hat{\underline{x}}_k$. The control is

$$\underline{u}_k = - \Lambda_{k+1} \Phi_{k+1,k} \hat{\underline{x}}_k \qquad 2.5.2$$

The control matrix Λ_{k+1} that results from the minimization of 2.5.1 is

$$\Lambda_{k+1} = (\Gamma^T_{k+1,k} \Pi'_{k+1} \Gamma_{k+1,k} + W^U_{k-1})^{-1} \Gamma^T_{k+1,k} \Pi'_{k+1} \qquad 2.5.3$$

$$\Pi'_{k+1} = \Phi^T_{k+2,k+1} \Pi_{k+2} \Phi_{k+2,k+1} + W^X_{k+1} \qquad 2.5.4$$

$$\Pi_{k+1} = \Pi'_{k+1} - \Pi'_{k+1} \Gamma_{k+1,k} \Lambda_{k+1} \qquad 2.5.5$$

The similarity between the control equations and the Kalman filter equations has been referred to as a "Duality Principle."

The matrix Π_{N-k} is the dual of the error covariance matrix P_k for an N-stage control policy. It provides one of the important measures of the behavior of the control system. However, the primary consideration of the evaluation of the linear guidance law resides in its ability to satisfactorily meet the original constraints imposed upon the trajectory in the specification of the mission. Since the deviation from the nominal is the basic criterion for evaluating the performance of the guidance system, the covariance of this deviation is computed from

$$M_k \overset{Df}{=} E[\underline{x}_k \underline{x}_k^T]$$

$$= \Phi_{k,k-1}(I - \Gamma_{k,k-1}\Lambda_k)(M_{k-1} - P_{k-1})(I - \Gamma_{k,k-1}\Lambda_k)^T$$

$$\Phi_{k,k-1}^T + \Phi_{k,k-1}P_{k-1}\Phi_{k,k-1}^T + Q_{k-1} \qquad 2.5.6$$

Thus, the P_k and M_k provide the basic statistical measures of the performance of the navigation and guidance systems.

3.0 COMPUTER PROGRAM DESCRIPTION

3.1 INTRODUCTORY AND EXPLANATORY REMARKS

Flow charts provide the basic framework around which the discussion which follows is constructed. These diagrams serve to indicate the logical flow connecting different functional blocks. As mentioned previously, these flow charts coupled with the identification of the subroutine to the block number contained in the Operator and Programmers Guide may be used in conjunction with the program listing to describe this program to its smallest detail.

3.1.1 Schema for Flow Chart Presentation

As has already been stated, the flow charts are arranged according to "levels." In the resulting hierarchy, the Level I flow chart provides the most general description since it depicts the overall program. Each functional block is further described by lower level flow charts. These charts indicate the logical flow within the block and describe the input and output requirements of the block. The equations used to obtain the desired outputs are presented as a supplement to the lowest level flow chart. The number of levels that are required depends upon the logical complexity of the functional block.

LEVEL I: This flow chart is designed to provide a very general description of the entire program. The titles assigned to the functional blocks are intended to be suggestive of the nature of the role to be performed within the block. Those functions that are to be performed in the basic computational cycle are designated by Roman numerals. Arabic symbols are used for functions that occur only once or play a passive role.

In a less complex program the input and output quantities required by the program could be described on this flow chart. However, this approach proved to be impracticable for this program so these requirements are described in the appropriately named functional blocks.

To indicate the basic logical decisions that can regulate and alter the flow between functional blocks, decision blocks are indicated. These decisions represent in a general manner the types of decisions that are required. The actual decision logic is described in the Level II flow charts of the functional blocks immediately preceding the decision block.

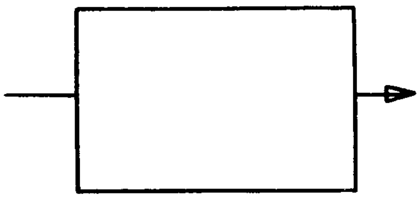
LEVEL II: The Level II flow charts provide the first concrete description of the program. Only the most important logical flow within each functional block is indicated on these diagrams. The quantities that are required for all logical and computational operations within this block are stated on this chart. These quantities are differentiated as being either INPUT (i.e., values provided initially by the engineer) or

COMPUTED (i.e., values determined in other portions of the program). The quantities that are required in other parts of the program, either for print-out or for computations, are also indicated on this flow chart. The functional blocks that appear on these diagrams are denoted by two symbols (e.g., II. 1 when discussing the "first" block in the Level II flow chart of functional block II) and a name. The names have been selected to provide some insight into the nature of the block.

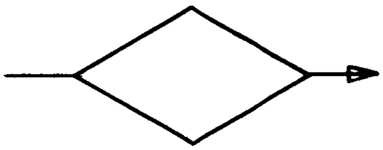
LEVEL III (and below): These diagrams provide additional details of the logical flow within the functional blocks depicted at Level II. These flow diagrams are augmented by the equations programmed into the computer. The input and output requirements of these blocks are stated on the diagrams. All of these quantities are summarized on the Level II flow chart.

3.1.2 Definition of Flow Chart Symbols

The following symbols represent the only ones that are used in the flow charts presented below.

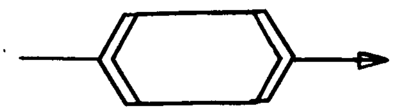


Set of operations that is to be described further by additional flow charts or by equations



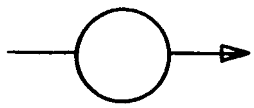
Logical decision

Operations that are predefined (i.e., in some other document)

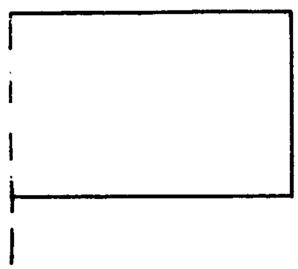


Operations that are completely defined by the statements contained within the box

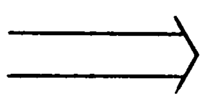
Connector used on Level II flow charts to indicate entry source and exit destination



Connector used on Level III flow charts



Summary of all quantities required in computations of flow charts on which this symbol appears or, alternatively, summary of all quantities computed in this flow chart which are required in other operations



This broad arrow appears on Level I and Level II flow charts. It is used to indicate information flow from one block to another. This symbol has been introduced to emphasize that many quantities are transmitted between the functional blocks in the higher level charts.

3.1.3 Definition of Mathematical Symbols

Subscripts

X_o	X is evaluated at $t = t_o$ (initial time)
X_c	X is evaluated at $t = t_c$ (control time)
X_G	X is evaluated at $t = t_G$ (nominal control time)
X_k	X is evaluated at $t = t_k$ (actual observation time)
X_p	X is evaluated at $t = t_p$ (minimum observation time)
X_P	X is evaluated at $t = t_P$ (print time)
${}_iX$	i refers to the i^{th} sensor

	1	ground tracker # 1
	2	ground tracker # 2
	3	ground tracker # 3
i =	4	horizon sensor
	6	radio altimeter
	7	inertial measurement unit

$X_{x, y, \text{ or } z}$	The x, y, or z component of X

Supercripts

s	Computed in spherical coordinates

Notation

$\underline{X}$	X is a vector
X^*	The nominal value of X
X^T	The transpose of X
X^{-1}	The inverse of X
X (ixj)	X is a matrix consisting of i rows and j columns
Data Rank	A designation of the "level" of the output data. When Rank 1 output is called for, all data having that rank is output. When Rank 2 is output is called for, both Rank 1 and Rank 2 data is output.

The symbols used in the subsequent flow charts and equations are defined below. These symbols appear in three groups: flags, Roman letter symbols, and Greek letter symbols. The dimensions are given in parenthesis following the definition. M denotes a dimension of mass, L a dimension of length, and T a dimension of time. If no designation is given, the quantity is unitless, and an R indicates an angular measure in radians. The dimensions of the diagonal elements of input Matrices are specified. The dimensions of the off-diagonal elements can be deduced from these.

FLAGS

BSFG (input quantity) Bias Error Flag. Indicates that constant random errors are included in the model of the observation process. This flag does not effect the IMU model in any way.

$$\text{BSFG} = \begin{matrix} 0, \text{ no bias errors} \\ 1, \text{ bias errors} \end{matrix}$$

DIMFG This flag consists of two numbers, m and n, which specify the dimensions of the observation matrix, ${}_A H$, the augmented state vector, ${}_A \underline{x}_k$, etc. Its value is determined in the initialization of the navigation block as a function of the instrument and bias flags.

HSFG (input quantity) Horizon Sensor Flag. On of four instrument flags which specifies whether or not the instrument is to be used as a source of measurement data.

$$\text{HSFG} = \begin{matrix} 0, \text{ no horizon sensor} \\ 1, \text{ use horizon sensor} \end{matrix}$$

IMFG (input quantity) Inertial Measurement Unit Flag. One of four instrument flags which specifies whether or not the instrument is to be used as a source of measurement data.

$$\text{IMFG} = \begin{matrix} 0, \text{ no IMU} \\ 1, \text{ use IMU} \end{matrix}$$

RAFG (input quantity) Radio Altimeter Flag. One of four instrument flags which specifies whether or not the instrument is to be used as a source of measurement data.

$$\text{RAFG} = \begin{matrix} 0, \text{ no radio altimeter} \\ 1, \text{ use radio altimeter} \end{matrix}$$

TRACC (input quantity) Constant altitude control flag. This flag specifies the manner in which the program switches to constant altitude control (phases 2 and 6)

0 - program switches to constant altitude control when $\dot{r} = 0$

1 - program switches to constant altitude control when

$$\ddot{r} < C_{apc}\, g_o \text{ and } \dot{r} > C_{vpc} \sqrt{g_o R}$$

where C_{apc} and C_{vpc} are input quantities. The program will always switch to constant altitude control on $r = 0$ if $r = 0$ before $\ddot{r} < C_{apc}\, g_o$ or $\dot{r} > C_{vpc} \sqrt{g_o R}$

TRBAK End of run flag. This flag is set by the program to determine the point at which a run is to be terminated.

TRCR1
TRCR2
TRCR3
TRCR4 A group of four flags used in the nominal trajectory block. They are used to specify which of four times (NEXTTi) has the minimum value. When TRCRi = 1, then NEXTTi is the minimum of the four values although all four or a lesser number may be minimum simultaneously.

TRFG (input quantity) Ground Tracker Flag. One of four instrument flags which specifies whether or not the instrument is to be used as a source of measurement.

TRFG =
0, no ground tracking
1, one ground tracker
2, two ground trackers
3, three ground trackers

TRGLM (input quantity) Guidance Law Flag. This flag determines whether a new tape will be generated by the Guidance Law Matrix Block.

TRGLM =
0, no new tape
1, make new tape

TRGUID (input quantity) Nominal Control Flag. This flag specifies whether a new roll angle comand, φ_c, is to be computed in the nominal trajectory block.

TRGUID
0, no new command
1, compute new φ_c

TRIMU (input quantity) IMU Tape Flag. This flag determines whether a new tape will be generated by the IMU error block

$$\text{TRIMU} = \begin{matrix} 0, \text{ no new tape} \\ 1, \text{ make new tape} \end{matrix}$$

TRINP (input quantity) Coordinate system type flag.

0 - initial position and velocity are input in spherical components $(r_o, \lambda_o, \mu_o, V_o, \gamma_o, A_o)$.

1 - initial position and velocity are input in cartesian components $(X_o, Y_o, Z_o, \dot{X}_o, \dot{Y}_o, \dot{Z}_o)$.

TRNIB (input quantity) Instrument Bias Error Flag. If TRNIB = 0, the bias errors are input. If TRNIB = 1, the bias errors are computed with a noise generator and the ${}_iB_O$ (i = 1, 2, 3, 4, 6, 7) matrices as covariances.

TRNIC (input quantity) Initial Condition Flag. This flag specifies whether gaussian noise or input quantities are to be added to the nominal initial conditions for use as initial conditions in the Actual Trajectory Block.

$$\text{TRNIC} = \begin{matrix} 0, \text{ input data} \\ 1, \text{ add noise} \end{matrix}$$

TRNOM (input quantity) Run Start Flag. This flag specifies where the run is to be commenced.

1, Start at Block I,
2, Start at Block III. Either or both TRGLM and TRIMU = 1
3, Start at Block IV
4, Start at tape edit, routine

TROMG (input quantity) Strapdown System Flag. This flag indicates whether the IMU is a strapdown or an inertial platform system.

$$\text{TROMG} = \begin{matrix} 0, \text{ inertial platform} \\ 1, \text{ strapdown system} \end{matrix}$$

This flag is required whenever a new IMU tape is to be generated.

TROPGN (input quantity) Time-varying gains computation flag. The control gains used in the constant altitude phases are computed as a function of time if this flag is set.

0 - input gains as constants $(K_{11}, K_{12}, K_{21}, K_{22})$

1 - compute gains as a function of time. Input damping ratio (ζ_1, ζ_2) and oscillation period (τ_1, τ_2)

TRPHSE (input quantity) Mission phase flag. The value of this flag corresponds to the phase of the nominal trajectory that the vehicle is currently in. This is an input quantity and the program may be started in any phase.

1 - first supercircular velocity phase

2 - first constant altitude phase

3 - skipout control phase

4 - free-fall phase

5 - second supercircular velocity phase

6 - second constant altitude phase

7 - subcircular velocity phase

TRPNT Print Flag. This flag specifies if data should be stored during the current iteration.

$$\text{TRPNT} = \begin{array}{l} 0,\ \text{do not store} \\ 1,\ \text{store Rank 1 data on tape 1} \\ 2,\ \text{store Rank 2 data on tape 1} \end{array}$$

TRSBCL (input quantity) Start subcircular velocity phase (7).

0 - if the program is presently in phase 6, then phase 7 will start when $r \leq 0$ and $|\varphi| \leq 10^{-2}$

1 - if the program is in phase 6, then phase 7 will start when $V = V_{IN}$

TRSKIN Skip integration flag. This flag is used in the program to avoid the possibility of integrating "backwards"

TRSTP (input quantity) Run Stop Flag. This flag specifies where the run is to be ended.

1, stop after completing nominal trajectory
2, stop after completing Guidance Law and IMU Error Matrices
3, stop after completing performance assessment runs and the tape edit

${}_o\zeta$	Navigation Flag. When ${}_o\zeta$ is zero at t_k, the Navigation Block is not used.
${}_i\zeta$	Instrument Flag. ζ refers to the i^{th} instrument.

$i =$

1	ground tracker # 1
2	ground tracker # 2
3	ground tracker # 3
4	horizon sensor
6	radio altimeter
7	IMU

at each t_k, when ${}_i\zeta$ is zero, the i^{th} instrument is not used.

Constants and Variables

$\underline{a}$	Vehicular acceleration. This vector has components $\ddot{X}$, $\ddot{Y}$, $\ddot{Z}$ along the $\underline{i}$, $\underline{j}$, $\underline{k}$ axes, respectively. (LT^{-2})
a	Magnitude of $\underline{a}$. (LT^{-2})
a'	Aerodynamic acceleration of vehicle in Earth g's.
a_e	Semi-major axes of a two-body conic trajectory calculated in phase 4. (L)
A_o	(input quantity) Initial azimuth of vehicle (part of initial velocity input). Used only when TRINP = 0. (R)
A_2	Orthonormal matrix transformation relating the body axes at $t = t_o$ $(\underline{P}_I, \underline{Y}_A, \underline{R}_O)_{t=t_o}$ to the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system.
A_3	Orthonormal matrix transformation defining the reference body axis $(\underline{P}_{Io}, \underline{Y}_{Ao}, \underline{R}_{Oo})$ in terms of the initial body axes $(\underline{P}_I, \underline{Y}_A, \underline{R}_O)_{t=t_o}$.
A_4	Orthonormal matrix transformation relating the reference body axes $(\underline{P}_{Io}, \underline{Y}_{Ao}, \underline{R}_{Oo})$ to the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system.
A_5	Orthonormal matrix transformation relating an initial local coordinate system $(\underline{i}_t, \underline{j}_t, \underline{k}_t)$ to the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system.
A_6	A (6x6) matrix defining the transformation which relates incremental quantities in the spherical coordinate system $(r, \theta, \phi, V, \gamma, \beta)$ in which the state transition matrix is computed to incremental quantities in the cartesian coordinate system in which the equations of motion are integrated.

${}_i a_j^1, {}_i b$ (input quantitiies, i = 1,2,3, j = 0,1,2,3) Range rate variance contants where $\sigma^2_{i\rho} \overset{Df}{=} {}_i a_o + {}_i a_1 (1 + {}_i b_i \rho)^2 {}_i\rho + {}_i a_2 (1 + {}_i b_i \rho)^2 \dot{}^2 + {}_i a_3 (1 + {}_i b_i \rho)^4 \dot{}$

${}_i b_j$ (input quantities, i = 1,2,3, j = 0,1,2) Range variance constants where $\sigma^2_{i\dot{\rho}} \overset{Df}{=} {}_i b_o + {}_i b_1 {}_i\rho^2 + {}_i b_2 {}_i\rho^4$

$\underline{b}$ (input quantity) The vector whose components are bias errors in the measurement data. The number of components is a function of the aiding instruments used and the BSFG flag. These quantities are input if TRNIB = 0 or computed using a noise generator if TRNIB = 1. (The dimensions of the components are specified under the definitions of the subvectors ${}_i\underline{d}$, ${}_i\underline{\alpha}$, $\underline{\epsilon}$)

${}_i B_{oo}$ (input quantity, i = 1,2,3,4,6) Diagonality flag for ${}_i B_o$ (or ${}_i B_I$ for i = 1,2,3). Value of zero indicates that ${}_i B_o$ is a diagonal matrix.

${}_i B_{oL}$ (input quantity, i = 1,2,3) Diagonality flag for ${}_i B_L$. Value of zero indicates that ${}_i B_L$ is diagonal matrix.

${}_7 B_{oGj}$ (input quantity, j = 1,2,3,4) Diagonality flag for ${}_7 B_{Gj}$. Value of zero indicates that ${}_7 B_{Gj}$ is a diagonal matrix.

${}_i B_o$ (input quantity, i = 1,2,3) Covariance of bias errors in tracker # i. This (7x7) matrix can be partitioned into the form

$$ {}_i B_o = \begin{bmatrix} {}_i B_I & 0 \\ 0 & {}_i B_L \end{bmatrix} $$

where ${}_i B_I$ is (3x3) and represents the covariance matrix of tracker location uncertainty. Elements of ${}_i B_I$ are ordered ${}_i x_T$, ${}_i y_T$, ${}_i z_T$ (i.e., the components of the position vector to the tracker at $t = t_o$). ${}_i B_L$ is (4x4) and represents the covariance matrix of bias errors in the tracker measurements. ${}_i B_L$ are ordered in terms of ${}_i\rho$, ${}_i\dot{\rho}$, ${}_i\psi$, ${}_i\eta$. (Dimensions of the diagonal elements are L^2, L^2, L^2, L^2, L^2T^{-2}, R^2, R^2.)

$_4B_o$ (input quantity) Covariance of the bias errors in horizon sensor. This is a (3x3) symmetric matrix. The elements of $_4B_o$ are ordered in terms of $\alpha, \delta, \beta H$ (Dimensions of diagonal elements are (R^2, R^2, R^2)

$_5B_o$ Covariance of bias error in space sextant.

$_6B_o$ (input quantity) Covariance of bias errors in the radio altimeter. This is a (2x2) symmetric matrix whose elements are ordered in terms of r and $\dot{r}$. (Dimensions of diagonal elements are L^2, L^2T^{-2})

$_7B_o$ (input quantity) Covariance of bias errors in the IMU. This (15x15) symmetric matrix is partitioned into the form

$$_7B_o = \begin{bmatrix} _7B_{G1} & 0 & 0 & 0 \\ 0 & _7B_{G2} & 0 & 0 \\ 0 & 0 & _7B_{G3} & 0 \\ 0 & 0 & 0 & _7BG_4 \end{bmatrix}$$

where $_7B_{Gj}$ (j = 1, 2, 3) are (3x3) symmetric matrices representing the covariance of the bias errors in gyro 1, 2, 3, respectively. $_7BG4$ is a (6x6) symmetric matrix representing the covariance of the bias errors in the three accelerometers. (Depends on K_i' i = 1, 2, ... 5. See User's Guide)

$B_{k,k-1}$ Constant parameter perturbation matrix. Relates constant random errors in lift and drag coefficients to the perturbation state vector. $B_{k,k-1}$ is a (6x2) matrix.

C_{apc} (input quantity) A quantity used to form $\ddot{r}_{pc}$ ($\ddot{r}_{pc} = C_{apc}\, g_o$) which is compared to radial acceleration. Used only if TRACC = 1. If $\ddot{r}_{pc} \geq \ddot{r}$ then a test is made on r to see if the program should switch from phase 1 to 2 or from phase 5 to phase 6. If $\ddot{r}_{pc} < \ddot{r}$, the program will switch to phases 2 or 6 when $\dot{r} = 0$.

C_{vpc} (input quantity) A quantity used to form r_{pc} ($\dot{r}_{pc} = C_{vpc}\sqrt{g_o\ R}$) used only if TRACC = 1. If $\ddot{r}_{pc} \geq \ddot{r}$ and $\dot{r}_{pc} \leq \dot{r}$, then the program switches to phases 2 or 6.

C_D Drag force ($\underline{D}$) coefficient as a function of α (angle of attack) ($C_D = C_{Do} + C_2\alpha^2 + C_4\alpha^4$).

C_{Do}, C_2, C_4 — (input quantities) Coefficients of C_D. In the nominal trajectory block $C_{Do} = C_{Do}^*$. In the actual trajectory $C_{Do} = C_{Do}^* + \delta C_{Do}$.

C_N — Normal force ($\underline{N}$) coefficient as a function of α (angle of attack) $(C_N = C_{N\alpha}\alpha + C_3\alpha^3 + C_5\alpha^5)$.

$C_{N\alpha}, C_3, C_5$ — (input quantities) Coefficients of C_N. In the nominal trajectory block $C_{N\alpha} = C_{N\alpha}^*$. In the actual trajectory $C_{N\alpha} = C_{N\alpha}^* + \delta C_{N\alpha}$.

C_{e1} — (input quantity) A constant used to calculate vehicle radiative heating (q_r). $C_e = C_{e1}$ when $(V/\sqrt{g\ r}) < 1.73$. $(M\ L^{-1-q_1}\ T^{q_1-3})$.

C_{e2} — (input quantity) A constant used to calculate vehicle radiative heating (q_r). $C_e = C_{e2}$ when $(V/\sqrt{g\ r}) \geq 1.73$. $(M\ L^{-1-q_2-3})$.

C_e — A constant used to calculate radiative vehicle heating (q_r). For velocity dependence and units see C_{e1} and C_{e2}.

C_H — (input quantity) A constant used to calculate stagnation point convective heating rate, q_c. Its value depends on the planetary atmosphere and the type of boundary layer flow.

$(M\ L^{1/2}\ T^{-3}$; e.g., English units $\rightarrow$ BTU $ft^{-3/2}\ sec^{-1})$

$C_{k,k-1}$ — Variable parameter perturbation matrix. Relates atmospheric density error to perturbation state vector. $C_{k,k-1}$ is (6x1) matrix.

$\underline{c}'$ — (input quantity) A (6x1) vector used to offset the nominal terminal state. $(L, L, L, LT^{-1}, LT^{-1}, LT^{-1})$

${}_iC_j$ — (input quantity, $i = 1,2,3$, $j = 1,2,\ldots,10$) Covariance matrix constant; multiplies computed covariance matrix of i^{th} ground tracker. As many as 10 of these can be input as a tabular function of time.

C_o — A (3x3) matrix defining the initial orientation of the body axes (P_{Io}, Y_{Ao}, R_{Oo}) with respect to the inertial reference system. This is used only in the IMU error section.

$C_{\zeta\tau 1}, C_{\zeta\tau 2}$ — Constants used to generate a commanded roll angle in the constant altitude control phase (2 and 6). $MT^{-1}, T^{-1})$

${}_i\underline{d}$ — (input quantity, $i = 1,2,3$) These are (7x1) vectors which describe the constant random bias error associated with ground trackers. They are subvectors of $\underline{b}$. If TRNIB = 1, these values are obtained from a noise generator. If TRNIB = 0, they are input directly $(L, L, L, L, LT^{-1}, R, R)$.

$\underline{D}$	Drag force. The aerodynamic force in the direction of negative velocity ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). ($M L T^{-2}$)
D	Magnitude of drag force. ($M L T^{-2}$)
e	Eccentricity of elliptical path in free-fall phase (4).
E_i	(input quantities, $i = 0, 1, 2, 3, 4$) Coefficients of fourth order polynominal in a' defining the maximum time a pilot can remain usefully conscious.
E_n	Integral of the ratio of the time a pilot spent at various acceleration levels to his maximum time of useful consciousness at those levels. When $E_n > 1$, the pilot has exceeded this tolerance level.
$\dot{E}_n$	Reciprical of time interval that a pilot can remain usefully conscious at a particular acceleration level. (T^{-1})
$\underline{f}$	Aerodynamic acceleration vector of the vehicle ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). (LT^{-2})
f	Magnitude of aerodynamic acceleration of the vehicle (LT^{-2})
F_i	($i = 0, 1, 2$) Constants defining the desired roll angle during the skipout control phases (3 and 3 modified). (The dimensions of F_o. F_1, F_2 are none, T^{-1}, T^{-2} respectively.)
F_{1i}	(input quantities, $i = 0, 1, 2$) F_i has these values during $t_3 \leq t < t'_3$ (phase 3). (unitless, T^{-1}, T^{-2})
F_{2i}	(input quantities, $i = 0, 1, 2$) F_i has these values during $t'_3 \leq t < t_4$ (phase 3 modified). (unitless, T^{-1}, T^{-2})
g	Vehicular acceleration due to graviational attraction of the re-entry planet ($g = g_o (R/r)^2$). (LT^{-2})
g_e	(input quantity) Sea level gravitational acceleration of earth. (LT^{-2})
g_o	(input quantity) Sea level gravitational acceleration of re-entry planet. (LT^{-2})
G_{max}	(input quantity) Limit on aerodynamic deceleration of the vehicle in earth g's. If this limit is exceeded during the actual trajectory run is terminated.

$\int_{t_o}^{t_k} G\,dt$ — A (3x15) matrix whose elements are velocity errors corresponding to those caused by bias errors in the IMU. $(LT^{-1}, LT^{-1}, LT^{-1})$

h — Altitude above surface of re-entry planet (planet assumed to be spherical). (L)

h_o — (input quantity) Input constant used to specify the covariance of noise on the atmospheric density perturbation (L)

h_ρ — (input quantity) Correlation altitude. Used in the calculation of the variance of the perturbation to atmospheric density. (L)

${}_iH_k$ — (i = 1,2,3,4,6,7) Observation matrix for i^{th} aiding instrument.

I (X) — A functional notation meaning the integer part of X.

$\underline{i}, \underline{j}, \underline{k}$ — An irrotational right-handed coordinate system of unit vectors. $\underline{i}$ and $\underline{j}$ are in the equatorial plane and $\underline{k}$ is along the polar axis. $\underline{i}$ is oriented so that the $\underline{i}$ $\underline{k}$ plane is the zero longitude meridian. Integration is performed in this cartesian coordinate system.

${}_aJ_k$ — State vector observation matrix for IMU.

${}_2J_k$ — Constant parameter observation matrix for IMU.

${}_3J_k$ — Variable parameter observation matrix for IMU.

k_H — (input quantity) Used in the calculation of radiative vehicle heating.

K_1, K_2 — Constant altitude guidance gains. Used to generate a commanded roll angle such that the vehicle will remain at a constant altitude during phases 2 and +. $(L^{-1}T, L^{-1})$

K_3 — Used to make the commanded roll angle have a transient value of π at the beginning of phases 2 and 6. If $K_3 > 10$, the transient is not applied. (T^{-1})

K_{11}, K_{12}, K_{13} — (input quantities) K_1, K_2, and K_3 have these values in phase 2. $(L^{-1}T, L^{-1}, T^{-1})$

K_{21}, K_{22}, K_{23} — (input quantities) K_1, K_2, and K_3 have these values in phase 6. $(L^{-1}T, L^{-1}, T^{-1})$

K_i' (i = 1,2,3,4,5) Scaling constants for IMU error sources. They consist, in the given order, of constant gyro readout error, random gyro drift, acceleration dependent drift, acceleration bias, and acceleration scale factor.

${}_iK_k$ (i = 1,2,3,4,6,7) The optimal gain matrix used during sequential processing.

${}_AK_k$ The augmented optimal gain matrix.

K_φ (input quantity) Pseudo autopilot gain. K_φ times the difference between the commanded roll angle and the present roll angle is the angular rate at which the vehicle will roll (up to a limit - see β_φ). (T^{-1})

k_i (input quantities, i = 0,1,2,3) These constants are used to calculate the variance of $\delta\rho_o$ at $t = t_p$ in the actual trajectory block in such a fashion that the uncertainty in the atmospheric model may be expressed as a function of altitude. ($ML^{-4}T$, ($ML^{-4}T$, $ML^{-5}T$, L^{-1})

M (input quantity) Mass of the vehicle. The mass is assumed constant throughout re-entry ignoring ablation effects. (M)

m (input quantity) An exponent used in the calculation of convective heating rate (q_c) at the stagnation point. For laminar flow m = 3 corresponding to a gas with viscosity proportional to the square root of temperature.

M_{oo} (input quantity) Diagonality of M_o. Value of zero indicates that M_o is diagaonal.

M_o (input quantity) Covariance of perturbation state vector. This is used to generate the deviation of the actual trajectory from the nominal at time $t = t_o$ if TRNIC = 1. (6x6). (Dimensions of the diagonal elements are L^2, L^2, L^2, L^2T^{-2}, L^2T^{-2}, L^2T^{-2})

M_c Covariance of perturbation state vector. (6x6) matrix associated with $\underline{x}_{kc}$.

${}_aM_c$ Covariance of augmented perturbation state vector. (9x9) matrix associated with ${}_A\underline{x}_{kc}$.

M_{IMU} (3x3) matrix defining orientation of instrument package (X_1, Y_1, Z_1) relative to initial pitch, yaw, roll axes.

n	(input quantity) An exponent used in the calculation of convective heating rate (q_c) at the stagnation point. Laminar flow is described by n = 1/2.
$\underline{N}$	Normal aerodynamic force. The aerodynamic force on the vehicle which is normal to the drag force ($\underline{D}$). The orientation of this vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) is determined by the roll angle (φ). (MLT^{-2})
N	Magnitude of $\underline{N}$ (normal aerodynamic force). (MLT^{-2})
N_{SP}	A program flag set to print the appropriate special condition number when the program stops on a special condition. See the User's Guide for a description of special condition program halts.
NEXTTi	A set (i = 1, 2, 3, 4) of program variables which are set equal to various significant program times. NEXTT1 = next time at which nominal control will be calculated NEXTT2 = next time at which a printout is called for NEXTT3 = next time at which a phase change will occur NEXTT4 = time at which a special condition has been encountered unless NEXTT4 = T_{END}
p	Semi-latus rectum of elliptical path of vehicle in free-fall phase (4). (L)
p_H	(input quantity) An exponent used to calculate radiative vehicle heating.
$\underline{P}$	Unit vector in direction of the pericenter (i, $\underline{j}$, $\underline{k}$ coordinates).
$\underline{P}_I, \underline{Y}_A, \underline{R}_O$	Orthonormal right-handed set of unit vectors along the pitch, yaw, and roll axes, respectively.
$\underline{P}_{Io}, \underline{Y}_{Ao}, \underline{R}_{Oo}$	Reference for body axes. These are not the same as $\underline{P}_I, \underline{Y}_A, \underline{R}_O$ at $t = t_o$ unless $\alpha_{10} = \alpha_{20} = \alpha_{30} = 0$.
P_k	Covariance of the error in the estimate of the perturbation state vector at time $t = t_k$. (6x6) matrix associated with $\underline{x}_k$.
${}_aP_k$	Covariance in estimate ${}_a\hat{\underline{x}}_k$ of ${}_a\underline{x}_k$. (9x9)
${}_aP'_k$	Covariance in estimate ${}_a\hat{\underline{x}}'_k$ of ${}_a\underline{x}'_k$. (9x9)
${}_AP_k$	Covariance in estimate ${}_A\hat{\underline{x}}_k$ of ${}_A\underline{x}_k$. (nxn)

P_o	Covariance of errors in the estimate of the perturbation state vector ${}_a\underline{x}_o$, at time $t = t_o$. (6x6) In the initialization block P_o is set equal to M_o.
${}_aP_o$	(input quantity) Covariance of the error in the estimate of the augmented perturbation state vector at $t = t_c$. (9x9) matrix associated with ${}_a\underline{x}_o$. ${}_aP_o$ may be partitioned as shown below.

$$ {}_aP_o = \begin{bmatrix} P_o & 0 & 0 \\ 0 & {}_1P_o & 0 \\ 0 & 0 & {}_2P_o \end{bmatrix} $$

${}_1P_o$	(input quantity) Covariance in bias errors in the drag coefficients at time $t = t_o$. (2x2) The elements are ordered in terms of δC_{Do} and $\delta C_{N\alpha}$.
${}_1P_{oo}$	(input quantity) Diagonality of ${}_1P_o$. Value of zero indicates that ${}_1P_o$ is a diagonal matrix.
${}_2P_o$	(input quantity) Covariance in bias errors in the sea level density coefficient at time $t = t_o$. (1x1) A diagonality index is not required for a scalar. (M^2L^{-6})
${}_3P_o$	(input quantity) Covariance in bias errors in the control system noise. This matrix consists of all zeros. (3x3)
q_1	(input quantity) An exponent used to calculate radiative vehicle heating (q_r). $q = q_1$ when $(V/\sqrt{gr}) < 1.73$.
q_2	(input quantity) An exponent used to calculate radiative vehicle heating (q_r). $q = q_2$ when $(V/\sqrt{gr}) \geq 1.73$.
q	An exponent used to calculate radiative vehicle heating (q_r). For velocity dependence see q_1 and q_2.
q_c	Convective heating rate per unit area at the stagnation point. $(M\ T^{-3})$
q_r	Radiative heating rate per unit area at the stagnation point. $(M\ T^{-3})$
q_s	Total heating rate per unit area at the stagnation point $(q_c + q_r)$. $(M\ T^{-3})$

Q_H — The time integral of the total vehicle heating rate per unit area at the stagnation point (q_s). Q_H is proportional to the total heat absorbed by the vehicle. (M T^{-2})

Q — (input quantity) A (3x3) convariance matrix which may be partitioned into the (2x2) covariance matrix of noise on the control variables, ${}_1Q$, and the (1x1) covariance matrix of noise in the atmospheric density model ${}_2Q$, as shown below.

$$Q \triangleq \begin{bmatrix} {}_1Q & 0 \\ 0 & {}_2Q \end{bmatrix}$$

${}_1Q$ — (input quantity) A (2x2) symmetric submatrix of Q representing the covariance of errors in the control variables $\delta\alpha$ and $\delta\rho$ respectively. This matrix is input with time as the argument in a table with 10 entries.

${}_1Q_{oo}$ — (input quantity) Diagonality of ${}_1Q$. Value of zero indicates that ${}_1Q$ is diagonal.

${}_2Q$ — A (1x1) submatrix of Q which is computed at $t = t_p$ in the actual trajectory block.

$\underline{r}$ — Current position vector of vehicle ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). (L)

$\underline{r}_t$ — Current position vector of vehicle in an initial coordinate system set up at $t = t_o$, ($\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$ coordinates). (L)

r_o — (input quantity) Initial ($t = t_o$) radial distance of vehicle from center of re-entry planet. When TRINP = 0, r_o is input as part of the initial position coordinates. (L)

$\underline{r}$ — Current position vector of vehicle ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). (L)

r_a — Magnitude of $\underline{r}_a$. (L)

r_c — (input quantity) Radial distance of the vehicle from the center of the re-entry planet at the beginning of phases 2 or 6. This quantity must be input only if the program is started in phases 2 or 6. (L)

r_{ma} — (input quantity) If r_a (apocenter distance) > r_{ma} in phase 4, the run will end. For earth re-entry, this value would probably be set equal to the radial distance of the lowest Van Allen radiation belt from the center of the earth. (L)

r_p Pericenter distance of the vacuum trajectory defined by position and velocity at the beginning phases 1 and 5. (L)

r_s (input quantity) Radial distance of vehicle from the center of the re-entry planet defining the beginning and end of phase 4. (L)

$\dot{r}$ Current value of radial speed of vehicle. (LT^{-1})

$\dot{r}_{pc}$ Used to test radial velocity for switching to phases 2 or 6 if TRACC = 1. The value of $\dot{r}_{pc}$ is determined by the input quantity C_{vpc} ($\dot{r}_{pc} = C_{vpc} \sqrt{g_o R}$). (LT^{-1})

r_o, λ_o, μ_o (input quantities) Initial position. Used only when TRINP = 0 and constitutes the initial position of the vehicle in spherical coordinates where r_o, λ_o, μ_o refer, respectively, to radial distance, latitude, and longitude. (L, R, R)

${}_i r_T, {}_i\varphi, {}_i\theta$ (i = 1, 2, 3) Spherical components of i^{th} tracker position. They are in turn radial distance to center of planet, longitude, and latitude.

r, λ, μ Current values of r_o, λ_o, μ_o. (L, R, R)

R (input quantity) Sea level radius of re-entry planet. (L)

R_N (input quantity) Radius of curvature of the vehicle at the heat stagnation point. (L)

$\underline{R}_O, \underline{R}_{Oo}$ See $\underline{P}_I$, $\underline{Y}_A$, $\underline{R}_O$ and $\underline{P}_{Io}$, $\underline{Y}_{Ao}$, $\underline{R}_{Oo}$, respectively.

${}_iR_o$ (input quantity, i = 1, 2, 3) Diagonality flag for ${}_iR_k$.

$${}_iR_o = \begin{cases} 0, & {}_iR_k \text{ is diagonal} \\ 1, & {}_iR_k \text{ is nondiagonal} \end{cases}$$

${}_iR_k$ (input quantities, i = 1, 2, 3) Covariance matrix of noise in ground trackers.

$${}_iR_k \stackrel{Df}{=} \begin{bmatrix} \sigma^2_{i\rho} & \sigma_{{}_i\rho\,{}_i\rho} & \sigma_{{}_i\rho\,{}_i\Psi} & \sigma_{{}_i\rho\,{}_i\eta} \\ & \text{Symmetric} & \sigma^2_{{}_i\Psi} & \sigma_{{}_i\Psi\,{}_i\eta} \\ & & & \sigma^2_{i\eta} \end{bmatrix}$$

$\sigma^2_{i\rho}$ and $\sigma^2_{i\dot\rho}$ are calculated using ${}_ia_j$, ${}_ib$, and ${}_ib_j$. The remainder of the elements are input directly. (The dimensions of the diagonal elements are L^2, L^2T^{-2}, R^2, R^2)

${}_4R_o$ (input quantity) Diagonality flag for ${}_4R_k$.

$${}_4R_o = \begin{cases} 0, & {}_4R_k \text{ is diagonal} \\ 1, & {}_4R_k \text{ is nondiagonal} \end{cases}$$

${}_4R_k$ (input quantity) Covariance of noise in horizon sensor measurements. This is a (3x3) matrix of which 6 elements are input as a tabulated function of time and 25 entries are possible. First 3 elements are diagonal elements. Ordered in terms of α, δ, βH.

${}_6R_o$ (input quantity) Diagonality flag for ${}_6R_k$.

$${}_6R_k = \begin{cases} 0, & {}_6R_k \text{ is diagonal} \\ 1, & {}_6R_k \text{ is nondiagonal} \end{cases}$$

${}_6R_k$ (input quantity) Covariance of noise in radio altimeter measurements. This is a (2x2) matrix of which 3 elements are input as a tabulated function of time. First 2 elements are diagonal elements. Ordered in terms of h, $\dot{h}$. (The dimensions of the diagonal elements are L^2, L^2T^{-2})

${}_7R_o$ (input quantity) Diagonality flag for ${}_7R_k$

$${}_7R_k = \begin{cases} 0, & {}_7R_k \text{ is diagonal} \\ 1, & {}_7R_k \text{ is nondiagonal} \end{cases}$$

${}_7R_k$ (input quantity) Covariance of noise in the IMU measurement. This is a (3x3) matrix of which 6 elements are input as a tabulated function of time. (The dimensions of the diagonal elements are L^2T^{-2}, L^2T^{-2}, L^2T^{-2})

S (input quantity) Aerodynamic area. The cross sectional aerodynamic area of the vehicle used to compute the aerodynamic forces.

t Current value of time (used in equations). (T)

t_i Current value of time (used in computer). (T)

t_{i+1} Value of time at the next cycle through the dynamic blocks. (T)

t_j ($j = G, p, k, c, P, W$) These are sets of time prints calculated within the program by means of the input T_j and Δt_j ($j = G, p, k, c, P$). The functions described below are accomplished at these times

t_G – nominal control is calculated.

t_p – possible observation time. Linear system matrices are evaluated and stored on tape at these time points

t_k – actual observation time point. Observations with aiding instruments are made and navigation is performed.

t_c – guidance time. Perturbative control quantities are computed at these time points.

t_P – save data time. Data is stored on the output tape during the performance assessment part of the program at these times.

t_W output print time. The tape edit routine prints the output at these times.

These time points form sets ($S(t_j) \triangleq$ set of t_j times) which must bear the following relation to each other.

$$S(t_c) \leq S(t_k) \leq S(t_p) \leq S(t_G)$$

and

$$S(t_W) \leq S(t_P) \leq S(t_G$$

T_{ji} ($j = G, p, k, c, P, W$) (input quantities, $i = 1, 2, \ldots, 10$) These are times defining the limits over which an interval of Δt_{ji} is used to specify the t_j time points defined above. The t_j times in the region $T_{ji-1} \leq t_j < T_{ji}$ are equally spaced with an interval of Δt_{ji} starting at T_{ji} and proceeding backwards in time. The T_{ji} must be integer multiples of Δt_{ji}. In addition to the above the t_G and t_p times occur at $t = t_o$ and phase changes.

t_o (input quantity) Initial time. Value of time at which program begins. (T)

t_1 Time at which the first supercircular phase beings (phase 1). (T)

t_2 Time at which the first supercircular phase ends and the first constant altitude phase (phase 2) begins. (T)

t_3 (input quantity) Time at which the first constant altitude phase ends and the first skipout control phase (phase 3) begins. (T)

t_3' (input quantity) Time at which the first skipout control phase ends and the second skipout control phase (phase 3 modified) begins. (T)

t_4 Time at which skipout control ends and the free-fall phase (phase 4) begins. (T)

t_5 Time at which the free-fall phase ends and the second supercircular phase (phase 5) begins. (T)

t_6 Time at which the second supercircular phase ends and the second constant altitude control phase (phase 6) begins. (T)

t_7 Time at which the second constant altitude phase ends and the subcircular phase (phase 7) begins. (T)

t_8 Time at which the subcircular phase ends (the program ends here also). (T)

t_{END} (input quantity) End time. The program will end when $t_i \geq t_{END}$. (T)

T_c (input quantity) Time at the beginning of the constant altitude control phases (2 and 5). This number must be input only if the program is started in either phase 2 or phase 5. (T)

T_c' (input quantity) Time at the beginning of either of the skipout control phases (phase 3 or phase 3 modified). This number must be input only if the program is started in either of the skipout control phases. (T)

If $t_3' \leq t < t_3'$ then $T_c' = t_3$

If $t_3' \leq t < t_4$ then $T_c' = t_3'$

TPLUS The minimum value of future time at which at least one of the following conditions is met. (T)

1. Nominal control calculation
2. Print and/or store data
3. Change phase
4. End run

(TPLUS is the time to which the program integrates — Block I.4.) (T)

$\underline{U}_p$	Unit vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) perpendicular to the current orbit plane.
$\underline{U}_{po}$	Unit vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) perpendicular to the initial orbit plane. The direction of $\underline{U}_p$ at $t = t_o$.
$\underline{U}_u$	Unit vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) in the orbit plane perpendicular to the current velocity vector.
$\underline{U}_r$	Unit vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) in the direction of the current position vector of the vehicle.
$\underline{U}_v$	Unit vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) in the direction of the current velocity vector of the vehicle.
$\underline{u}_c$	A (2x1) optimal control perturbation vector which is added to the nominal control vector. The nominal control consists of constant input angle of attack, α, commanded roll angle φ_c. (R, R)
$\underline{V}_a$	Velocity ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates) at the apocenter calculated at the beginning of phase 4. (LT^{-1})
V_a	Magnitude at $\underline{V}_a$. (LT^{-1})
V_o, γ_o, A_o	(input quantities) Initial velocity in spherical coordinates. Input only if TRINP = 0. The coordinates are respectively initial speed, initial flight path angle, and initial azimuth of the vehicle. (LT^{-1}, R, R)
V, γ, A	Current values of V_o, γ_o, and A_o. (LT^{-1}, R, R)
$\underline{V}$	Velocity vector ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). (LT^{-1})
$\underline{V}_t$	Velocity vector ($\underline{i}_t$, $\underline{j}_t$, $\underline{k}_t$ coordinates set up at $t = t_o$). (LT^{-1})
V_{IN}	(input quantity) If the program is in phase 6 and TRSBCL = 1, phase 7 will begin at $V = V_{IN}$. Also, if the program is in phase 3 (or 3 modified) and $\dot{r} \leq 0$ and $V \leq V_{IN}$, the program ends. (LT^{-1})
V_N	Performance index for optimal guidance and navigation.
W_{co}^U	(input quantity) Diagonality flag for W_c^U. $W_{co} = 0 \Longrightarrow$ diagonal matrix; $W_{co} = 1 \Longrightarrow$ nondiagonal matrix.
W_c^U	(input quantity) A symmetric (2x2) control weighting matrix whose elements are tabulated functions of 10 time points. Three elements are given with the diagonals listed first. (The dimensions of the diagonal elements are R^{-2}, R^{-2})

W_{co}^X (input quantity) Diagonality flag for W_c^X. $W_{co}^X = 0 \Rightarrow$ diagonal matrix; $W_{co}^X = 1 \Rightarrow$ nondiagonal matrix.

W_c^X (input quantity) Terminal constraint weighting matrix. It is a symmetric (9x9) matrix whose elements are tabulated against 10 time points. 21 elements per time point are required because the matrix is partitioned as shown below.

$$W^X \triangleq \begin{bmatrix} W' & 0 \\ 0 & 0 \end{bmatrix}$$

X, Y, Z Current position components of the vehicle. The direction of these magnitudes is along the $\underline{i}$, $\underline{j}$, $\underline{k}$ unit vectors, respectively. (L, L, L)

$\dot{X}, \dot{Y}, \dot{Z}$ Current velocity components of the vehicle ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system). (LT^{-1}, LT^{-1}, LT^{-1})

$\ddot{X}, \ddot{Y}, \ddot{Z}$ Acceleration components of the vehicle ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system). (LT^{-2}, LT^{-2}, LT^{-2})

X_o, Y_o, Z_o (input quantities) Initial position components (at $t = t_o$) of vehicle in the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system. Used only if TRINP = 1. (L, L, L)

$\dot{X}_o, \dot{Y}_o, \dot{Z}_o$ (input quantities) Initial velocity components (at $t = t_o$) of vehicle in the $\underline{i}$, $\underline{j}$, $\underline{k}$ coordinate system. Used only if TRINP = 1. (LT^{-1}, LT^{-1}, LT^{-1})

X_a, Y_a, Z_a Position components of vehicle at apocenter. This position is calculated at the beginning of phase 4 and represents the maximum distance the vehicle will achieve from the center of the re-entry planet ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). (L, L, L)

$\dot{X}_a, \dot{Y}_a, \dot{Z}_a$ Velocity components of vehicle at apocenter. This velocity is calculated at the beginning of phase 4 and represents the velocity of the vehicle at its maximum distance from the planet ($\underline{i}$, $\underline{j}$, $\underline{k}$ coordinates). (LT^{-1}, LT^{-1}, LT^{-1})

$\underline{X}_k$ The 6-dimensional state vector. The first 3 components denote position and the last 3 represent velocity. The nominal state is written $\underline{X}_k^*$.

$$\underline{X}_k \triangleq \begin{bmatrix} \bar{r} \\ \bar{V} \end{bmatrix}$$

$\underline{x}_k$ The 6-dimensional perturbation state vector.

$$\underline{x}_k \triangleq \underline{X}_k - \underline{X}^*_k$$

$\underline{x}_{Dif}$ The position and velocity difference between the actual trajectory and nominal trajectory i.e. $\underline{x}_{Dif} = \underline{X} - \underline{X}^*$ (6x1)

$\underline{\tilde{x}}_k$ The error in the estimate i.e. $\underline{\tilde{x}}_k = \underline{\hat{x}}_k - \underline{x}_{Dif}$ (6x1)

$\underline{\hat{x}}'_k$ Best linear estimate of $\underline{x}_k$ based on measurement data $\underline{z}_{k-1}$, (6x1)

${}_a\underline{x}_o$ (input quantity) ${}_a\underline{x}_k$ at $t = t_o$. IF TRNIC = 1, ${}_a\underline{x}_o$ is obtained from a noise generator in the initialization. If TRNIC = 0 it is input. (9x1)

$\underline{Y}_A$, $\underline{Y}_{Ao}$ See $\underline{P}_I$, $\underline{Y}_A$, $\underline{R}_O$ and $\underline{P}_{Io}$, $\underline{Y}_{Ao}$, $\underline{R}_{Oo}$ respectively.

${}_i\underline{Y}_k$ (i = 1,2,3,4,6,7) The observation vector with components consisting of measurements that would be made by the ith instrument at time $t = t_k$. These consist of:

i	Instrument	Observation	Dimensions
1	1st ground tracker	${}_1\rho$, ${}_1\dot{\rho}$, ${}_1\psi$, ${}_1\eta$	L, LT^{-1}, R, R
2	2nd ground tracker	${}_2\rho$, ${}_2\dot{\rho}$, ${}_2\psi$, ${}_2\eta$	L, LT^{-1}, R, R
3	3rd ground tracker	${}_3\rho$, ${}_3\dot{\rho}$, ${}_3\psi$, ${}_3\eta$	L, LT^{-1}, R, R
4	horizon sensor	α, δ, B^H	R, R, R
6	radio altimeter	h, $\dot{r}$	L, LT^{-1}
7	IMU	$\int_{t_o}^{t_k} \underline{f}\, dt$	LT^{-1}, LT^{-1}, LT^{-1}

${}_i\underline{y}_k$ (i = 1,2,3,4,6,7) The perturbative observation vector of the ith instrument

$${}_i\underline{y}_k = {}_i\underline{Y}_k - {}_i\underline{Y}^*_k$$

${}_i\underline{z}_k$	($i = 1,2,3,4,6,7$) The perturbed observation vector of the i^{th} instrument degraded by instrument bias errors and noise on the observation. The dimensions are the same as the ${}_i\underline{y}_k$)
α	Angle of attack. This is the angle between R_O (the roll axis fixed in the vehicle) and the velocity vector ($\underline{V}$) and is assumed to remain constant throughout the nominal trajectory. α is equal to α' during phase 1, 2, and 3; α'' during phase 4, 5, 6, and 7. (R)
α'	(input quantity) The angle of attack (α) has the value α' during phases 1, 2, and 3. (R)
α''	(input quantity) The angle of attack (α) has the value α'' during phases 4, 5, 6, and 7. (R)
$\alpha_1, \alpha_2, \alpha_3$	Body Euler angles. These angles define the present orientation of the body fixed axes ($\underline{P}_I$, $\underline{Y}_A$, $\underline{R}_O$) with respect to the position of the reference body fixed axes ($\underline{P}_{I0}$, $\underline{Y}_{A0}$, $\underline{R}_{O0}$). The transformation is shown in Block I. 5. (R, R, R)
$\alpha_{10}, \alpha_{20}, \alpha_{30}$	(input quantities) Initial values (at $t = t_o$) of α_1, α_2, α_3. The transformation is shown in Block B4. (R, R, R)
${}_i\underline{\alpha}$	($i = 4,6$) Bias errors on the aiding instruments. These are constant random variables related to the following instruments. i = 4 (3x1) horizon sensor (R, R, R) i = 6 (2x1) radio altimeter (L, LT^{-1})
β	Angle of the velocity vector projected onto the local horizontal.
β'	(input quantity) Re-entry planet atmospheric density decay factor. (L^{-1})
β_φ	(input quantity) Limit on the roll angle rate. Regardless of the commanded roll angle, the vehicle will not exceed a roll rate of β_φ. (T^{-1})
$\beta^H_{min}, \beta^H_{max}$	(input quantities) Minimum and maximum possible subtended angles for horizon sensor. (R, R)
γ_k	Control measurement matrix. (6x2)
γ_o	(input quantity) Input only if TRINP = 1. Initial value (at $t = t_o$) of γ. (R)

γ_{max}	(input quantity) Maximum value γ may have at the beginning of phase 4 without the program terminating. (R)
γ_{min}	(input quantity) Minimum value γ may have at the beginning of phase 4 without the program terminating. (R)
$\Gamma_{k,k-1}$	Control perturbation matrix. Relates the control vector $\underline{u}_{k-1}$ to the perturbation state vector $\underline{x}_k$. $\Gamma_{k,k-1}$ is a (6x2) matrix.
${}_c\Gamma_{k,k-1}$	White noise weighting matrix in density perturbation shaping filter.
$\delta C_{Do}, \delta C_{N\alpha}$	(input quantities) Constant random variables describing the uncertainty in the drag and normal aerodynamic coefficients. These are the 7th and 8th components of the (9x1) augmented state vector ${}_a\underline{x}_k$. If TRNIC = 1, the noise generator is used to obtain these values for use in the actual trajectory block using, P_o as the covariance matrix. If TRNIC = 0. δC_{Do} and $\delta C_{N\alpha}$ are input.
δt	Integration step size used by the integration routine. (T)
δt_1	(input quantity) δt has the value δt_1 for all phases except phase 4. (T)
δt_2	(input quantity) δt has the value δt_2 for phase 4. (T)
δt_{IM}	(input quantity) The integration step size desired for the integration routine in the IMU Error Matrices section. The step size used is the minimum of δt_{IM} and the interval to the next t_G time point. (T)
$\delta \rho_o$	The density perturbation. During nominal trajectory $\delta \rho_o = 0$, but during the actual trajectory the density perturbation model is assumed to be $$\delta \dot{\rho}_o = - \frac{\lfloor \dot{h} \rfloor}{h_\rho} \delta \rho_o + w_\rho (t)$$
Δt_{ji}	(j = G, p, k, c, P, W) (input quantity, i = 1, 2, . . . , 10) Intervals used to specify t_j time points. (see t_j). (T)
Δh_{max}	(input quantity) Constant used as a trajectory constraint. The performance assessment run is terminated when the difference between altitudes in nominal and actual trajectories exceeds this value. (L)
ΔR_{max}	(input quantity) Constant used as a trajectory constraint. The performance assessment run is terminated when the distance between a point on the nominal trajectory and a corresponding point at the same time on the actual trajectory exceeds this value. (L)

${}_i\Delta y$	($i = 1,2,3,4,6,7$) Measure of the linearity of the observation of the i^{th} instrument. It is obtained by subtracting the perturbative observation of the i^{th} instrument, predicted on the basis of linear theory, from the difference of the observation as seen from the nominal and actual trajectory. (The dimensions are the same as the ${}_i\underline{y}_k$)
$\Delta_{k,k-1}$	Plant noise perturbation matrix. Relates white noise in plant equations to the state vector.
ϵ	The value of time, t, obtained within the integration routine must agree within ϵ to the time established as an integration routine exit time. (T)
ϵ_1	(input quantity) ϵ has the value ϵ_1 during all phases except phase 4. (T)
ϵ_2	(input quantity) ϵ has the value ϵ_2 during phase 4. (T)
ϵ_I	(input quantity) ϵ has value ϵ_I during the IMU Error Matrices calculation. (T)
ϵ_s	Yaw guidance constant. A number whose magnitude is less than 1 which is used to define yaw control limits.
$\underline{\epsilon}$	(input quantity) IMU error sources. A 15 component vector of constant random errors which is obtained from a noise generator if TRNIB = 1 or input if TRNIB = 0. (R, RT^{-1}, $RL^{-1}T^2$, R, RT^{-1}, $RL^{-1}T^2$, R, RT^{-1}, $RL^{-1}T^2$, LT^{-2}, LT^{-2}, LT^{-2}, dimensionless, dimensionless, dimensionless)
ζ	Damping ratio of constant altitude roll angle. $\zeta = \zeta_1$ in phase 2 and $\zeta = \zeta_2$ in phase 6. Input only if TROPGN = 1.
ζ_1, ζ_2	(input quantities) Damping ratios for constant altitude roll angle control Input only if TROPGN = 1. The damping ratio of the constant altitude roll angle control will be ζ_1 and ζ_2 for phases 2 and 6, respectively, if TROPGN = 1.
θ	Range angle plus 90^o. An angle measured in the $\underline{r}\ \underline{k}_t$ plane from $\underline{k}_t$ to r. (R)
${}_i\eta$	Azimuth angle measured by ground tracker. (R)
$\underline{\eta}_k$	Control system noise as measured by IMU at time $t = t_k$.

λ Latitude of vehicle. Angle between the $\underline{i}\ \underline{j}$ plane and present position vector. (R)

λ_o (input quantity) Initial latitude of vehicle. Input only if TRINP = 0. (R)

Λ_c Control gain matrix (2x9)

π_c Control cost matrix (9x9)

π'_c Extrapolated control cost matrix. (9x9)

ρ Atmospheric density of re-entry planet (ML^{-3})

ρ_o (input quantity) Sea level atmospheric density of re-entry planet. (ML^{-3})

${}_i\rho$ (i = 1, 2, 3) Range as measured by the ith ground tracker. (L)

${}_i\dot{\rho}$ (i = 1, 2, 3) Range rate as measured by the ith ground tracker. (LT^{-1})

ρ^1_{max} (input quantity) If the range from any tracker to the vehicle is greater than this constant, the variance in range measurement for that tracker is set to 10^6. (L)

ρ^2_{max} (input quantity) If the range from any tracker to the vehicle is greater than this constant, the variances for the angular measurements is set to 10^6. (L)

$\sigma^2_{i\psi}$ Variance of noise in measurement of elevation angle ${}_i\psi$. (R^2)

$\sigma^2_{i\eta}$ Variance of noise in measurement of azimuth angle ${}_i\eta$. (R^2)

$\sigma^2_{i\rho}, \sigma^2_{i\dot{\rho}}$ Covariance of noise in range ${}_i\rho$ and range rate ${}_i\dot{\rho}$ measurements. (L^2T^{-1})

$\sigma_{i\rho i\dot{\rho}}$ Covariance of noise in range ${}_i\rho$ and range rate ${}_i\dot{\rho}$ measurements. (L^2T^{-1}0

$\sigma_{i\rho i\psi}$ Covariance of noise in range ${}_i\rho$ and elevation angle ${}_i\psi$ measurements. (LR)

$\sigma_{i\rho i\eta}$ Covariance of noise in range ${}_i\rho$ and azimuth angle ${}_i\eta$ measurements. (LR)

$\sigma_{{}_i\dot{\rho}{}_i\psi}$ Covariance of noise in range rate ${}_i\dot{\rho}$ and elevation angle ${}_i\psi$ measurements. (LT)

$\sigma_{{}_i\dot{\rho}{}_i\eta}$ Covariance of noise in range rate ${}_i\dot{\rho}$ and azimuth angle ${}_i\eta$ measurements. (LR)

$\sigma_{{}_i\psi{}_i\eta}$ Covariance of noise in elevation angle ${}_i\psi$ and azimuth angle ${}_i\eta$ measurements. (R^2)

${}_i\sigma$ ($i = 1,2,3,4,6,7$) A priori statistics error. Constant that allows the effect of incorrect a priori statistics to be examined. $(1 + {}_i\sigma)\ {}_iR_k$ is used to generate noise vector ${}_i\underline{v}_k$ whereas ${}_iR_k$ is used to determine estimates.

σ_k Estimate of integrated control effect.

τ Period of constant altitude roll angle. $\tau = \tau_1$ in phase 2 and $\tau = \tau_2$ in phase 6. Used only if TROPGN = 1. (T)

$\tau'(a)$ The interval of time that a pilot can remain usefully conscious at a given acceleration (a) level. τ' is a function of a. (T)

τ_1, τ_2 (input quantities) Constant roll angle control periods. Input only if TROPGN = 1. The natural period of the constant altitude roll angle control will be τ_1 and τ_2 for phases 2 and 6, respectively, if TROPGN = 1. (T, T)

φ, φ_1 Current value of the roll angle. φ is the angle between the vertical plane through the velocity vector ($\underline{V}$) and the normal force ($\underline{N}$). (R)

φ_{i-1} The value of φ_i resulting from the previous nominal control calculation. (R)

φ_o (input quantity) Initial roll angle (φ at $t = t_o$). (R)

φ_c Commanded roll angle. (R)

φ_{c1} The value of φ_c during phases 1 and 5 (supercircular constant roll angle control phases). (R)

φ_{c3} (input quantity) The value of φ_c during phase 7 (subcircular constant roll angle control phase). (R)

φ_{11} (input quantity) The value of φ_{c1} during phase 1 (first supercircular constant roll angle control phase). (R)

φ_{21}	(input quantity) The value of φ_{c1} during phase 5 (second supercircular constant roll angle control phase). (R)
ϕ	An angle measured in the $\underline{i}_t$ $\underline{j}_t$ plane from $\underline{j}$ to the plane formed by $\underline{r}$ and $\underline{k}_t$. (R)
$\Phi(t, t_k)$	The (6x6) dimensional state transition matrix relating the states $\underline{x}(t)$ and $\underline{x}(t_k)$.
$_a\Phi(t, t_k)$	The (9x9) dimensional state transition matrix relating the states $_a\underline{x}(t)$ and $_a\underline{x}(t_k)$.
$_A\Phi(t, t_k)$	The (nxn) dimensional state transition matrix relating the states $_A\underline{x}(t)$ and $_A\underline{x}(t_k)$.
$_c\Phi(t, t_k)$	The (1x1) dimensional state transition matrix which is the solution of the linear homogeneous differential equation for the perturbative density function.
$\Psi(t, t_k)$	The (6x6) dimensional state transition matrix which is the solution to the adjoint linear differential equations.
$_i\psi$	($i = 1, 2, 3$) Elevation angle measured by the i^{th} tracker. (R, R, R)
$_i\psi_o$	(input quantity, $i = 1, 2, 3$) Minimum permissible elevation angle for the i^{th} tracker. (R, R, R)
$\omega_{\varphi i-1}$	Value used by program in the calculation of φ_c if $\lvert\omega_{\varphi i-1}\rvert \leq \beta_\varphi$. If $\lvert\omega_{\varphi i-1}\rvert > \beta_\varphi$, $\omega_{\varphi i-1}$ is replaced by β_φ. (T^{-1})
ω_{PI}, ω_{YA}, ω_{RO}	Components of angular rate (ω) along the pitch, yaw, and roll axes of the vehicle, respectively. (T^{-1}, T^{-1}, T^{-1})

3.2 FUNCTIONAL ORGANIZATION OF THE PROGRAM

The diagram that immediately follows these paragraphs is designated as the Level I flowchart. It does nothing other than summarize the basic structure of the program in terms of the basic functional operations that must be performed. It can be considered as consisting of two types of functions. First, operations that constitue the basic computational cycle; these functions are described by the blocks that have been designated with Roman numberals. The details relative to these blocks can be found in Section 4. Blocks A, B, and C describe functions that either occur once (i.e., Block B), are required in order to make the program operate meaningfully (i.e., Block A), or act passively relative to the computational cycle (i.e., Block C). These three blocks are described in Section 3.0.

The INPUT block represents a summary of the quantities that an engineer must input. No computations are contained within this block. In the GENERAL INITIALIZATION block, computations that must be performed once during a specific simulation run and/or logical decisions that must be made for proper operation within the basic computational cycle are accomplished. The OUTPUT block consists primarily of the output tape edit routine which presents the output of the program in a readily understood and usable form.

This program uses guidance and navigation policies that are based upon the techniques of linear perturbation theory. To apply these methods, it is necessary to compute a nominal (or reference) trajectory. This task is accomplished in the NOMINAL TRAJECTORY BLOCK.

This program is divided into three basic computational stages. These stages are, in order, the nominal (Blocks I and II), the guidance law and IMU error matrices (Block III), and the performance assessment (Blocks IV through VII) stages. A computer run may be started at the beginning of any of the three stages if the earlier stages, if any, have already been run and the data tape, which is generated with each of these stages, is available.

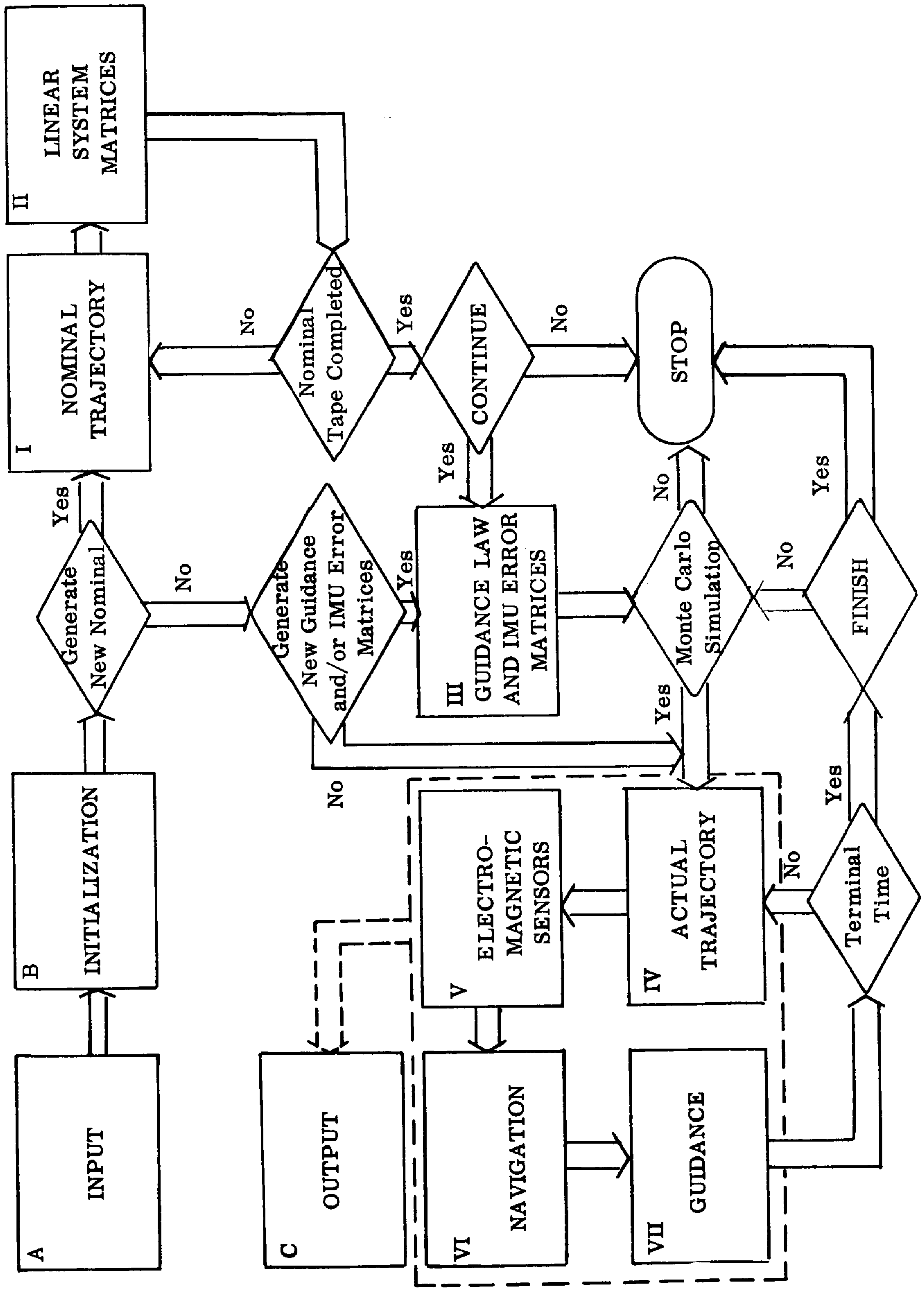
Certain characteristic types of time points are in use throughout the three stages. Associated with these time points are the time intervals between them. The nominal control time points, t_G, form a set consisting of the times at which a nominal control command is given to the re-entry vehicle. It has the shortest time interval of all the types of time points. The minimum observation interval is defined by the time points, t_p, and are a subset of the t_G. The actual observation times, t_k, are a subset of the t_p and the actual control times, t_c, are a subset of the t_k. Data output from the performance assessment stage occurs at times, t_P, which are a subset of the t_G.

The LINEAR SYSTEM MATRICES and the NOMINAL TRAJECTORY blocks are conceptually linked in the sense that both blocks depend only on the nominal trajectory which is being flown. For this reason, although the tape handling techniques for this program have not yet been defined, it is anticipated that the tape generated in these two blocks, containing all the pertinent and necessary data for calculations in subsequent blocks, will be distinct from tapes containing data which is influenced by the parameters of the study. The function of the LINEAR SYSTEM MATRICES BLOCK is to compute the state transition matrix, $\Phi(t_p, t_{p-1})$ and the other matrices which appear in the linear dynamical and IMU equations. Since the data in these blocks is independent of the rest of the program, the program is set up so that only these two blocks, along with the appropriate initialization, may be computed if desired. For the same reason, time progresses simultaneously in both blocks from $t = t_o$ to $t = t_N$ in two sets of intervals. These are the intervals defined by the t_G's and the t_p's.

The GUIDANCE LAW AND IMU ERROR MATRIX BLOCK consists of two sections which are combined for the sake of convenience. The output of the GUIDANCE LAW MATRIX section consists of matrices which are evaluated at $t = t_c$ starting at $t = t_N$ and continuing until $t = t_0$. These matrices are used in the GUIDANCE BLOCK in order to implement the guidance philosophy and are stored on tape until needed. The output of the IMU ERROR MATRIX section is also stored on tape for use in the NAVIGATION BLOCK. It is, however, stored in such a fashion that either it or the output of the GUIDANCE LAW MATRIX section can be changed independently, depending on the nature of the parametric study. The output of the second section consists of (3 x 15) matrices which are obtained by integrating the IMU caused errors in acceleration from t_0 to t_p and tabulated at every t_p.

Parametric studies of sensors, control intervals, observation times, etc., may be accomplished in the performance assessment part of the program. The program may be started at this point by using previously generated tapes if desired. The nominal initial conditions of the trajectory may be perturbed to form the initial conditions for the actual trajectory. If a non-zero estimate of the state is input, perturbative control is generated at this time. The equations of motion are integrated in the ACTUAL TRAJECTORY BLOCK until observation time points have been reached, at which time observations by the appropriate electromagnetic sensors in the ELECTROMAGNETIC SENSOR BLOCK, or observations by the IMU, are simulated.

Navigation is performed at every observation time in the NAVIGATION BLOCK; part of the output consists of the best estimate of the state as obtained from a Kalman filter. If the observation time point is also a guidance time point, this best estimate is used in the GUIDANCE BLOCK to generate an optimal perturbative control vector which is added to the nominal control until a new value is computed.



Level 1 Flow Chart - Guidance and Navigation for Aided-Inertial Re-entry

3.3 INPUT, GENERAL INITIALIZATION, AND OUTPUT

3.3.1 Input - Block A

The total input to this program is contained on 22 pages of load sheets which are contained in paragraph 4.10 in the User's Guide. This program is unit independent, with the exception of angular units which must be input as radians. If the input has been made in a consistent set of units, all calculations and output is made in that same set of units.

A listing of the input always precedes a computer run. If intermediate tapes are generated, the output tape edit routine presents the input, along with identifying alphanumeric symbols at the beginning of the run as well as at the beginning of the performance assessment part of the program. If only a performance assessment run is made, the input which was used to generate the intermediate tapes is still printed at the beginning of the run even though the card input to the parts of the program used to generate the intermediate tapes are not supplied. This is accomplished by reading this input information from the intermediate tapes where it has been stored in order that each run made by the program does contain all the input which was made to generate the data in that run.

The following pages consist first of the listing of the input for a typical run and then of the print of the input as made by the output tape edit routine.

```
SINGLE  9   1 DEC 1.,1.0,1.0,4.,5.,1.,1.,3.0     *  PRINT TAPE ONLY
SINGLE  9   1 DEC 1.,1.,1.,4.,1.,0.,0.,2.0      *    NOM ONLY
SINGLE  9   1 DEC 1., 1.0, 1.0, 4.,2.,0.,1.,3.0     *  GLM, PA
SINGLE  9   1 DEC 1., 1., 1., 4., 1., 0, 1., 3.0    *   NOM, GLM, AND PA
SINGLE  9   1 DEC 1.,1.0,1.0,4.,3.,0.,0.,3.0    *    PA ONLY
SINGLE  9   1 DEC 1.,1.0,1.0,4.,1.,1.,1.,3.0     *  NOM, IMU, GLM, AND PA
SINGLE  9   9 BCI ,EQUATORIAL ENTRY TRAJECTORY
SINGLE  9  19 BCI , STATE VECTOR REDUCED TO 6X1
SINGLE  1  11 DEC 0,7.,1.,0,0
SINGLE  1  16 DEC 6450.,0,0
SINGLE  1  19 DEC 8.,-.11,1.5707963
SINGLE  1  22 DEC 1500.
SINGLE  1  32 DEC 5.
SINGLE  1  42 DEC       0., 0., 0.     *  ALPHA*10  ALPHA*20  ALPHA*30
SINGLE  1  45 DEC 3.,.698,.7
SINGLE  1  50 DEC 0
SINGLE  1  61 DEC       0.5411, 0.3926991 *  ALPHAPPRIME  PHI*21               0013
SINGLE  1  70 DEC 6500.0703,123.,170.
SINGLE  1  73 DEC       0., 0.         *  C*V*PC  C*A*PC
SINGLE  1  75 DEC       1.5057098, -1.8170089, 0.92657443 *  C*D*0  C*2  C*4
SINGLE  1  78 DEC       1.3608741, -1.3843004, 0.29525989 *  C*N*ALPHA  C*3  C*5
SINGLE  1  81 DEC 4985.4930,.2286E-3,.12021652E-4
SINGLE  1  84 DEC       0.22485927E5, -0.99277738E4, 0.16798636E4 *  E*0  E*1  E*2
SINGLE  1  87 DEC       -0.12717953E3, 0.36059210E1, 1.98E4 *  E*3  E*4  C*H
SINGLE  1  89 DEC .37208550E10
SINGLE  1  96 DEC 3.,.5,.9806635E-2,.98066350E-2
SINGLE  1 100 DEC .13915225E10,.13961021,6378.165,100.
SINGLE  1 104 DEC 1500.
SINGLE  1 114 DEC 5.
SINGLE  1 124 DEC 0,120.,.3048
SINGLE  1 127 DEC .5,1.E-4
SINGLE  1 129 DEC 64.,1.E-3
SINGLE  2   1 DEC 0
SINGLE  2   2 DEC 1.,0,0,0,1.,0,0,0,1.
SINGLE  2  11 DEC .5,1.E-3
SINGLE  2  13 DEC .48481368E-5,.72921150E-7,.74358993E-5,.98066350E-8,1.E-6
SINGLE  2  18 BCI ,HEADER FOR IMU
SINGLE  3   1 DEC 1500.
SINGLE  3  11 DEC 15.
SINGLE  3  21 DEC 0.,0.
SINGLE  3  24 DEC 200.,1.E4,1.E4
SINGLE  3  24 DEC 200.,1.E3,1.E3
SINGLE  3  64 DEC 200.,1.,1.,1.,1.E6,1.E6,1.E6
SINGLE  3  64 DEC 110.,1.,1.,1.,1.E4,1.E4,1.E4
SINGLE  3  86 DEC 500.,1.E4,1.E4,1.E4,1.E4,1.E4,1.E4
SINGLE  3 285 BCI, GUIDANCE LAW MATRICES
SINGLE  4   1 DEC       1.          *  TRNIC
SINGLE  4   1 DEC 0
SINGLE  4   2 DEC .1,.1,0,.01,.01,0
SINGLE  4   8 DEC .1,.05,2.E4
SINGLE  4   8 DEC .1,.05                      *DELCD,DELCN
SINGLE  4   8 DEC 0,0                       *DELCD,DELCN
SINGLE  4  10 DEC 0
SINGLE  4  11 DEC 2.E14,0,0,0,0
```

```
SINGLE  4  11 DEC 0
SINGLE  4  16 DEC 20.,1.E6,3.,0,1.E6,1.E6
SINGLE  4  16 DEC 50.                       * GMAX
SINGLE  4  33 DEC 1.,1.,1.,.01,.01,.01
SINGLE  4  54 DEC   .01,.01        *  1PD
SINGLE  4  54 DEC 0,0
SINGLE  4  60 DEC 1500.
SINGLE  4  70 DEC 15.
SINGLE  4  80 DEC 1500.
SINGLE  4  90 DEC 15.
SINGLE  5   1 DEC 0,0,0,0,0,1.,0          *RA ONLY
SINGLE  5   8 DEC 1.E10,1.E10
SINGLE  5  10 DEC 6378.165,6378.165,6378.165
SINGLE  5  13 DEC 0,0,0                   *EQ TRAJ TRACKER LAT
SINGLE  5  13 DEC .1,.1,.1                 *TRACKER LAT
SINGLE  5  13 DEC 0,1.,-1.               *TRACKET LAT.
SINGLE  5  16 DEC .924,.925,.926
SINGLE  5  16 DEC .01,.1,.15           *  EQ TRAJ TRACKER LONG
SINGLE  5  16 DEC .01,1.,1.              *TRACKER LONG
SINGLE  5  19 DEC 0,0,0
SINGLE  5  19 DEC -2.,-2.,-2.          * VISIBILITY CRIT ELEV
SINGLE  5  22 DEC 10.,1.,1.,1.
SINGLE  5  62 DEC 0,0,0
SINGLE  5  65 DEC .9E-9,0,0,0,.9E-9,0,0,0,.9E-9,0,0,0
SINGLE  5  77 DEC 0,0,0
SINGLE  5  80 DEC .225E-3,0,0,.225E-3,0,0,.225E-3,0,0
SINGLE  5  91 DEC .30461742E-5,.30461742E-5
SINGLE  5 101 DEC .30461742E-5,.30461742E-5
SINGLE  5 111 DEC .30461742E-5,.30461742E-5
SINGLE  5 119 DEC 0,0,6.3,6378.165
SINGLE  5 123 DEC 10.,.30461742E-5,.30461742E-5,.30461742E-5
SINGLE  5 298 DEC 0
SINGLE  5 299 DEC 1500.,.9E-3,.25E-8
SINGLE  5 399 DEC 0
SINGLE  5 400 DEC 200.,900.E-10,900.E-10,900.E-10
SINGLE  6   1 DEC 1.,0
SINGLE  6   3 DEC -1.,-1.,-1.,-1.,-1.
SINGLE  6   3 DEC 0,0,0,0,0,0,0
SINGLE  6  10 DEC 0,0,0,0,0,0,0,0,0,0,0
SINGLE  6  22 DEC 1.11E-6,1.21E-6,1.31E-6
SINGLE  6  28 DEC .9E-3,.25E-8,.30461742E-5,.30461742E-5
SINGLE  6  38 DEC 1.12E-6,1.22E-6,1.32E-6
SINGLE  6  44 DEC .9E-3,.25E-8,.30461742E-5,.30461742E-5
SINGLE  6  54 DEC 1.13E-6,1.23E-6,1.33E-6
SINGLE  6  60 DEC .9E-3,.25E-8,.30461742E-5,.30461742E-5
SINGLE  6  70 DEC .30461742E-5,.30461742E-5,.30461742E-5
SINGLE  6  77 DEC .9E-3,.25E-8
SINGLE  6  80 DEC 0,4.,25.
SINGLE  6  86 DEC 4.,4.,25.
SINGLE  6  92 DEC 4.,4.,25.
SINGLE  6  98 DEC 100.,25.,100.,25.,100.,25.
SINGLE  6 120 DEC 0.,0.
SINGLE  6 119 DEC 1000.,1.E-4,1.E-4
SINGLE  6 159 DEC .1,.1,0,.01,.01,.0
```

```
RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT            RUN NO.   1  PAGE  57
STATE VECTOR REDUCED TO 6X1
NOMINAL TRAJECTORY AND LINEAR SYSTEM MATRICES INPUT
************************************************************************************************
   TRINP =   0     TRPHSE =   7     TRSBCL =   1     TROPGN =   0     TRACC =   0
TRAJECTORY DATA
        R*0             LAM*0             MU*0              V*0              GAM*0             A*0
   0.64500000F 04   0.00000000E-38   0.00000000E-38   0.80000000E 01   -0.11000000E 00   0.15707963E 01
   T*G NOMINAL CONTROL INTERVAL
   0.15000000E 04   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   DELT*G
   0.50000000E 01   C.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   C.00000000E-38   C.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
     ALPH*10          ALPH*20          ALPH*30           K*PHI           BTA*PHI           EPS*5
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.30000000E 01   0.69800000E 00   0.70000000E 00
       T*C              R*C             PHI*0            PHI*C3            V*IN
   C.00000000E-38   C.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
       K*11             K*12             K*13             ALPHP            PHI*11
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
       K*21             K*22             K*23             ALPHPP           PHI*21            T*CP
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.54110000E 00   0.39269910E 00   0.00000000E-38
       F*10             F*11             F*13             F*20             F*21              F*22
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
       R*S              T*3              T*3P             CVPC             CAPC
   0.65000703E 04   0.12300000E 03   0.17000000E 03   0.00000000E-38   0.00000000E-38
VEHICLE DATA
       C*D0             C*2              C*4             C*NALP             C*3              C*5
   0.15057098E 01  -0.18170089E 01   0.92657442E 00   0.13608741E 01   -0.13843004E 01   0.29525989E 00
        M               R*N               S
   0.49854930E 04   0.22860000E-03   0.12021652E-04
```

RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT RUN NO. 1 PAGE 58

STATE VECTOR REDUCED TO 6X1

NOMINAL TRAJECTORY AND LINEAR SYSTEM MATRICES INPUT

****

PHYSICAL ENVIRONMENT

E*0	E*1	E*2	E*3	E*4	C*H
0.22485927E 05	-0.99277737E 04	0.16798636E 04	-0.12717953E 03	0.36059210E 01	0.37208550E 10
C*E1	**C*E2**	**Q*1**	**Q*2**	**PH**	**KH**
0.00000000E-38	C.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38
M-EX	**N-EX**	**G*0**	**G*E**	**RHO*0**	**BETAP**
C.30000000E 01	0.50000000E 00	C.98066350E-02	0.98066350E-02	0.13915225E 10	0.13961021E 00
CAPR	**H*RHO**				
0.63781650E 04	0.10000000E 03				

PROGRAM CONTROL

T*P MINIMUM OBSERVATION INTERVAL

0.15000000E 04	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38
0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38

DELT*P

0.50000000E 01	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38
0.00000000E-38	C.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38

T*0	T*END	V*END	DELT1	EP11	DELT2	EP12
0.00000000E-38	0.12000000E C3	0.30480000E 00	0.50000000E 00	0.99999999E-04	0.64000000E 02	0.99999999E-03

RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT RUN NO. 1 PAGE 59

STATE VECTOR REDUCED TO 6X1

IMU MATRICES INPUT

```
*************************************************************************************************************

TROMG =   0     DELTIM =  0.50000000E 00      EPII =  0.99999999E-03

3 BY  3  DIAGONAL MATRIX  M*IMU  INSTRUMENT ORIENTATION MATRIX
 0.10000000E 01  0.10000000E 01  0.10000000E 01

        K*1P             K*2P             K*3P             K*4P             K*5P

   0.48481368E-05   0.72921149E-07   0.74358992E-05   0.98066349E-08   0.10000000E-05
```

```
RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT          RUN NO.    1  PAGE  60
STATE VECTOR REDUCED TO 6X1
                    GUIDANCE LAW MATRICES INPUT
****************************************************************************************************
T*C
 0.15000000E 04   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
 0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
DELT*C
 0.15000000E 02   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
 0.00000000E-38   C.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
WU*O =    0     WX*O =   0     TRGLM =   1
WCU(2X2) SYMMETRIC CONTROL WEIGHTING MATRICES
     DIAGONAL MATRIX
 TIME           ELE 11          ELE 22
0.20000000E 03  0.10000000E 04  0.10000000E 04
WP*CX(6X6) SYMMETRIC STATE WEIGHTING MATRICES
     DIAGONAL MATRIX
 TIME           ELE 11          ELE 22          ELE 33          ELE 44          ELE 55          ELE 66
0.11000000E 03  0.10000000E 01  0.10000000E 01  0.10000000E 01  0.10000000E 05  0.10000000E 05  0.10000000E 05
0.50000000E 03  0.10000000E 05  0.10000000E 05  0.10000000E 05  0.10000000E 05  0.10000000E 05  0.10000000E 05
```

```
RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT          RUN NO.    1  PAGE  61
STATE VECTOR REDUCED TO 6X1
                    PERFORMANCE ASSESSMENT INPUT
*******************************************************************************************
ACTUAL TRAJECTORY INPUT ****
   TRNIC =    0
      DELXO            DELYO            DELZO            DELXO.           DELYO.           DELZO.
   0.10000000E 00   0.10000000E 00   0.00000000E-38   0.99999999E-02   0.99999999E-02   0.00000000E-38
      DELCDO           DELCNA           DELRHO
   0.00000000E-38   0.00000000E-38   0.00000000E-38
      K*0              K*1              K*2              K*3              H*0
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
      G*MAX            R*MA             GAMAX            GAMIN            DELHM            DELRM
   0.50000000E 02   0.10000000E 07   0.30000000E 01   0.00000000E-38   0.10000000E 07   0.10000000E 07
   M*00 =    0      P*00 =    0      2P*00 =    0
   M*0 (6X6) COVARIANCE MATRIX OF INITIAL STATE
      DIAGONAL MATRIX
             ELE 11           ELE 22           ELE 33           ELE 44           ELE 55           ELE 66
          0.10000000E 01   0.10000000E 01   0.10000000E 01   0.99999999E-02   0.99999999E-02   0.99999999E-02
   P*0 (2X2) COVARIANCE MATRIX OF AERODYNAMIC COEFFICIENTS
      DIAGONAL MATRIX
             ELE 11           ELE 22
          0.00000000E-38   0.00000000E-38
   T*K OBSERVATION INTERVAL
   0.15000000E 04   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   DELT*K
   0.15000000E 02   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
```

```
RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT          RUN NO.   1  PAGE  62
STATE VECTOR REDUCED TO 6X1
                         PERFORMANCE ASSESSMENT INPUT
************************************************************************************************
  T*CAPP TAPE WRITE INTERVAL
   C.15000000E 04   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   0.00000000E-38   0.00000000E-38   C.00000000E-38   0.00000000E-38   0.00000000E-38
  DELT*CP
   0.15000000E 02   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38
   C.00000000E-38   C.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38

INPUT FOR NAVIGATION SYSTEM ****
   BSFG = 0     SSFG = 0     HSFG = 0     TRFG = 0     MINFG = 0
   RAFG = 1     IMFG = 0

RADIO ALTIMETER-
R6(0) =    0
        TIME              R6(11)            R6(22)
 T 1  0.15000000E 04    0.90000000E-03    0.25000000E-08
  TRNIB =   1       IQ00 =   0
      1SIG             2SIG             3SIG             4SIG             5SIG             6SIG             7SIG
   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   0.00000000E-38   C.00000000E-38   0.00000000E-38
      1B0I             2B0I             3B0I             1B0L             2B0L             3B0L
       0                0                0                0                0                0
      4B00             6B00             7B0G1            7B0G2            7B0G3            7B0G4
       0                0                0                0                0                0
  1BI (3X3) COVARIANCE OF TRACKER NO. 1 LOCATION UNCERTAINTY
      DIAGONAL MATRIX
              ELE 11           ELE 22           ELE 33
          C.11100000E-05   0.12100000E-05   0.13100000E-05
```

```
RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT          RUN NO.    1  PAGE  63
                    STATE VECTOR REDUCED TO 6X1
                              PERFORMANCE ASSESSMENT INPUT
*************************************************************************************************

  1BL (4X4) COVARIANCE OF TRACKER NO. 1 BIAS ERRORS
      DIAGONAL MATRIX
              ELE 11          ELE 22          ELE 33          ELE 44
          0.90000000E-03  0.25000000E-08  0.30461742E-05  0.30461742E-05
  2BI (3X3) COVARIANCE OF TRACKER NO. 2 LOCATION UNCERTAINTY
      DIAGONAL MATRIX
              ELE 11          ELE 22          ELE 33
          0.11200000E-05  0.12200000E-05  0.13200000E-05
  2BL (4X4) COVARIANCE OF TRACKER NO. 2 BIAS ERRORS
      DIAGONAL MATRIX
              ELE 11          ELE 22          ELE 33          ELE 44
          0.90000000E-03  0.25000000E-08  0.30461742E-05  0.30461742E-05
  3BI (3X3) COVARIANCE OF TRACKER NO. 3 LOCATION UNCERTAINTY
      DIAGONAL MATRIX
              ELE 11          ELE 22          ELE 33
          0.11300000E-05  0.12300000E-05  0.13300000E-05
  3BL (4X4) COVARIANCE OF TRACKER NO. 3 BIAS ERRORS
      DIAGONAL MATRIX
              ELE 11          ELE 22          ELE 33          ELE 44
          0.90000000E-03  0.25000000E-08  0.30461742E-05  0.30461742E-05
  4BO (3X3) COVARIANCE OF HORIZON SENSOR BIAS ERRORS
      DIAGONAL MATRIX
              ELE 11          ELE 22          ELE 33
          0.30461742E-05  0.30461742E-05  0.30461742E-05
  5BO (1X1) COVARIANCE OF SPACE SEXTANT BIAS ERRORS
      DIAGONAL MATRIX
```

```
                    RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT        RUN NO.   1  PAGE  66
                   STATE VECTOR REDUCED TO 6X1
                                 PERFORMANCE ASSESSMENT INPUT
**********************************************************************************************************
          ELE 11
        0.00000000E-38
 6B0 (2X2) COVARIANCE OF RADIO ALTIMETER BIAS ERRORS
    DIAGONAL MATRIX
          ELE 11          ELE 22
        0.90000000E-03  0.25000000E-08
 7BG1 (3X3) COVARIANCE OF GYRO 1 BIAS ERRORS
    DIAGONAL MATRIX
          ELE 11          ELE 22          ELE 33
        0.00000000E-38  0.40000000E 01  0.25000000E 02
 7BG2 (3X3) COVARIANCE OF GYRO 2 BIAS ERRORS
    DIAGONAL MATRIX
          ELE 11          ELE 22          ELE 33
        0.40000000E 01  0.40000000E 01  0.25000000E 02
 7BG3 (3X3) COVARIANCE OF GYRO 3 BIAS ERRORS
    DIAGONAL MATRIX
          ELE 11          ELE 22          ELE 33
        0.40000000E 01  0.40000000E 01  0.25000000E 02
 7BG4 (6X6) COVARIANCE OF ACCELEROMETER BIAS ERRORS
    DIAGONAL MATRIX
          ELE 11          ELE 22          ELE 33          ELE 44          ELE 55          ELE 66
        0.10000000E 03  0.25000000E 02  0.10000000E 03  0.25000000E 02  0.10000000E 03  0.25000000E 02
 1Q (2X2) COVARIANCE OF CONTROL SYSTEM NOISE
    DIAGONAL MATRIX
  TIME            ELE 11          ELE 22
0.10000000E 04  0.99999999E-04  0.99999999E-04
```

```
RE-ENTRY GUIDANCE AND NAVIGATION PERFORMANCE ASSESSMENT          RUN NO.    1  PAGE  65
STATE VECTOR REDUCED TO 6X1
                       PERFORMANCE ASSESSMENT INPUT
**********************************************************************************************
  ESTIMATE OF STATE - ASMXHAT
   0.10000000E 00  0.10000000E CO  0.00000000E-38  0.99999999E-02  0.99999999E-02  0.00000000E-38
   0.00000000F-38  0.00000000E-38  0.00000000E-38
  INSTPUMENT BIAS ERRORS - SMBBAR
   0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38

   0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38

   0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38
   0.00000000E-38  0.00000000E-38  0.00000000E-38
   0.00000000E-38
   0.00000000F-38  0.00000000E-38
   0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38
   0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38  0.00000000E-38
   0.00000000E-38
GUIDANCE INPUT ****
       OFFX              OFFY              OFFZ              OFFX.             OFFY.             OFFZ.
   0.00000000E-38    0.00000000E-38    0.00000000E-38    0.00000000E-38    0.00000000E-38    0.00000000E-38
```

3.3.2 General Initialization

3.3.2.1 Initilization Format

The initialization for the Nominal Trajectory Block and the Linear Systems Matrices Block is accomplished in Blocks B.1 and B.2 respectively.

The initialization for the IMU Error Matrices and Guidance Law Matrices is accomplished in Blocks B.3a and B.3 respectively.

The initialization for the performance assessment blocks; i.e., the Actual Trajectory, the Electromagnetic Sensor, the Navigation and the Guidance blocks is done in Blocks B.4, B.5, B.6, and B.7 respectively.

3.3.2.2 Detailed Flow Charts and Equations

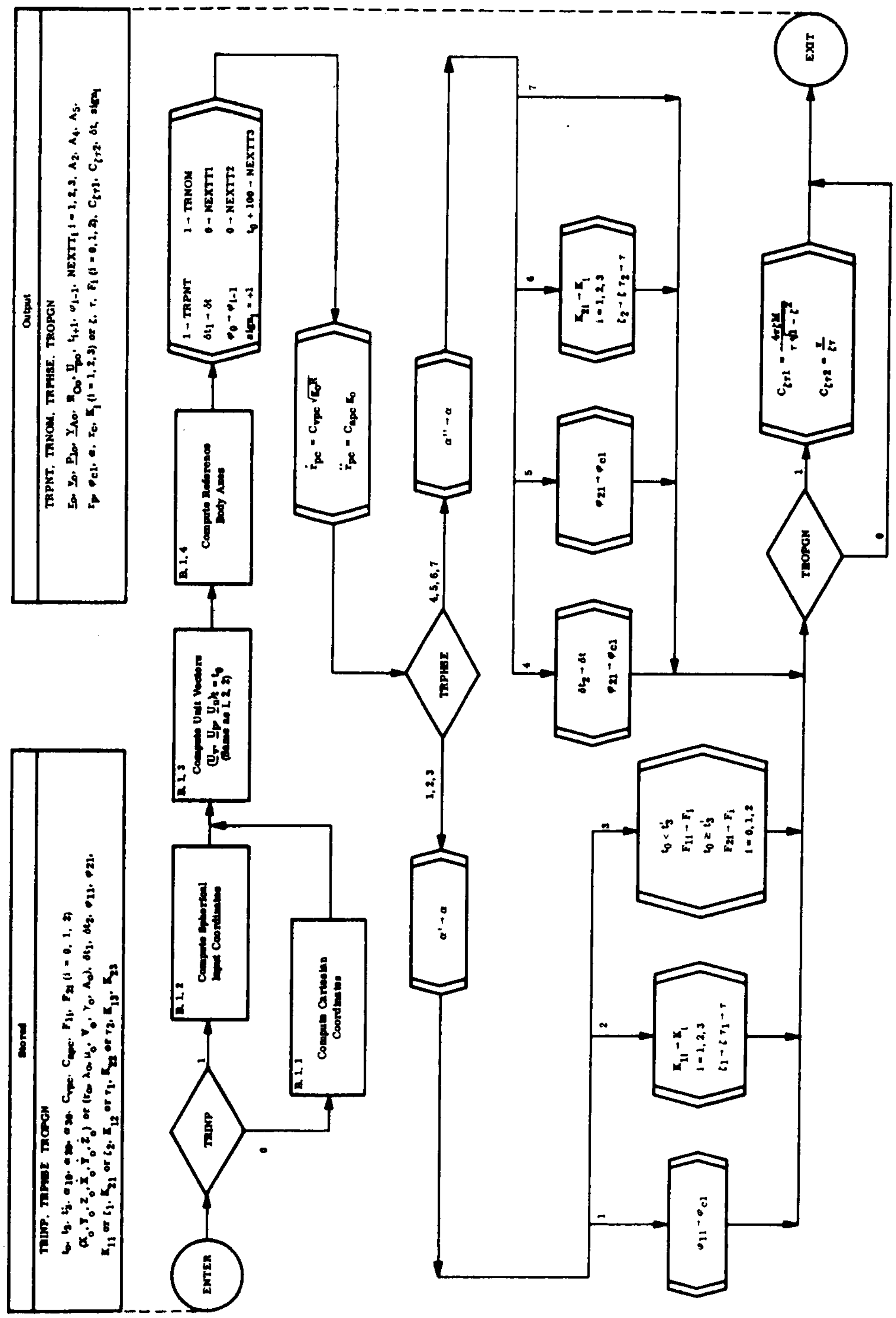


Figure 3.3.2.2.1 - Nominal Initialization Block B.1

Block B.1.1 Compute Cartesian Coordinates

INPUT: r_o, λ_o, μ_o, V_o, γ_o, A_o

OUTPUT: X_o, Y_o, Z_o, $\dot{X}_o$, $\dot{Y}_o$, $\dot{Z}_o$

$$X_o = r_o \cos \lambda_o \cos \mu_o$$

$$Y_o = r_o \cos \lambda_o \sin \mu_o$$

$$Z_o = r_o \sin \lambda_o$$

$$\dot{X}_o = V_o(- \cos \gamma_o \cos A_o \sin \lambda_o \cos \mu_o - \cos \gamma_o \sin A_o \sin \mu_o + \sin \gamma_o \cos \lambda_o \cos \mu_o)$$

$$\dot{Y}_o = V_o(- \cos \gamma_o \cos A_o \sin \lambda_o \sin \mu_o + \cos \gamma_o \sin A_o \cos \mu_o + \sin \gamma_o \cos \lambda_o \sin \mu_o)$$

$$\dot{Z}_o = V_o (\cos \gamma_o \cos A_o \cos \lambda_o + \sin \gamma_o \sin \lambda_o)$$

Block B. 1. 2 Compute Spherical Input Coordinates

INPUT: $X_o, Y_o, Z_o, \dot{X}_o, \dot{Y}_o, \dot{Z}_o$

OUTPUT: $r_o, \lambda_o, \mu_o, V_o, \gamma_o, A_o$

1. $$r_o = +\sqrt{X_o^2 + Y_o^2 + Z_o^2}$$

2. $$\mu_o = \tan^{-1}\left[\frac{Y_o}{X_o}\right] \qquad -\pi < \mu_o \leq \pi$$

3. $$\lambda_o = \sin^{-1}\left[\frac{Z_o}{r_o}\right] \qquad -\frac{\pi}{2} \leq \lambda_o \leq \frac{\pi}{2}$$

4. $$V_o = +\sqrt{\dot{X}_o^2 + \dot{Y}_o^2 + \dot{Z}_o^2}$$

5. $$\gamma_o = \sin^{-1}\left[\frac{X_o\dot{X}_o + Y_o\dot{Y}_o + Z_o\dot{Z}_o}{r_o V_o}\right] \qquad -\frac{\pi}{2} \leq \gamma_o \leq \frac{\pi}{2}$$

6. $$A_o = \tan^{-1}\left[\frac{-\sin\mu_o\dot{X}_o + \cos\mu_o\dot{Y}_o}{-\sin\gamma_o\cos\mu_o\dot{X}_o - \sin\lambda_o\sin\mu_o\dot{Y}_o + \cos\lambda_o\dot{Z}_o}\right]$$

$$-\pi < A_o \leq \pi$$

Block B. 1. 3 Compute Unit Vectors $(\underline{U}_v, \underline{U}_p, \underline{U}_u)_{t=t_o}$

This block is the same as I. 2. 2 (Compute Aerodynamic Forces) even though the only quantities needed are $\underline{U}_v$, $\underline{U}_p$, $\underline{U}_u$.

INPUT: $\underline{r}_o$, $\underline{v}_o$, ρ_o, β', R, S, C_{D_o}, C_2, C_4, C_{N_α}, C_3, C_5, α, φ_o

OUTPUT: $(\underline{U}_v, \underline{U}_r, \underline{U}_p, \underline{D}, \underline{N}, \dot{r})_{t=t_o}$

1. $V = +\sqrt{\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2}$
2. $\underline{U}_v = \dfrac{\underline{V}}{V}$
3. $r = +\sqrt{X^2 + Y^2 + Z^2}$
4. $\underline{U}_r = \dfrac{\underline{r}}{r}$
5. $\gamma = \sin^{-1} [\underline{U}_r \cdot \underline{U}_v]$
6. $\dot{r} = V \sin \gamma$
7. $\underline{U}_u = \dfrac{\underline{U}_r - \underline{U}_v \sin \gamma}{\cos \gamma}$
8. $\underline{U}_p = \underline{U}_u \times \underline{U}_v$
9. $\rho = \rho_o e^{-\beta'(r - R)}$
10. $C_D = C_{Do} + C_2 \alpha^2 + C_4 \alpha^4$
11. $C_N = C_{N\alpha} \alpha + C_3 \alpha^3 + C_5 \alpha^5$
12. $\underline{D} = - C_D \rho \dfrac{V^2 S}{2} \underline{U}_v$
13. $\underline{N} = C_N \rho \dfrac{V^2 S}{2} [\cos \varphi_i \underline{U}_u - \sin \varphi_i \underline{U}_p]$

NOTE: $(\underline{U}_p)_{t=t_o} = \underline{U}_{po}$, $\underline{U}_{po}$ is to be stored for later use.

Block B. 1. 4 Compute Reference Body Axes

INPUT: $(\underline{U}_v, \underline{U}_p, \underline{U}_u)_{t = t_o}$; α; α_{10}; α_{20}; α_{30}; φ_o; λ_o; μ_o; A_o

OUTPUT: $\underline{P}_{Io}$, $\underline{Y}_{Ao}$, $\underline{R}_{Oo}$, A_2, A_4, A_5

1. $\underline{U}_n \triangleq U_{mx}\underline{i} + U_{my}\underline{j} + U_{mz}\underline{k}$; $\quad m = v, p, u$

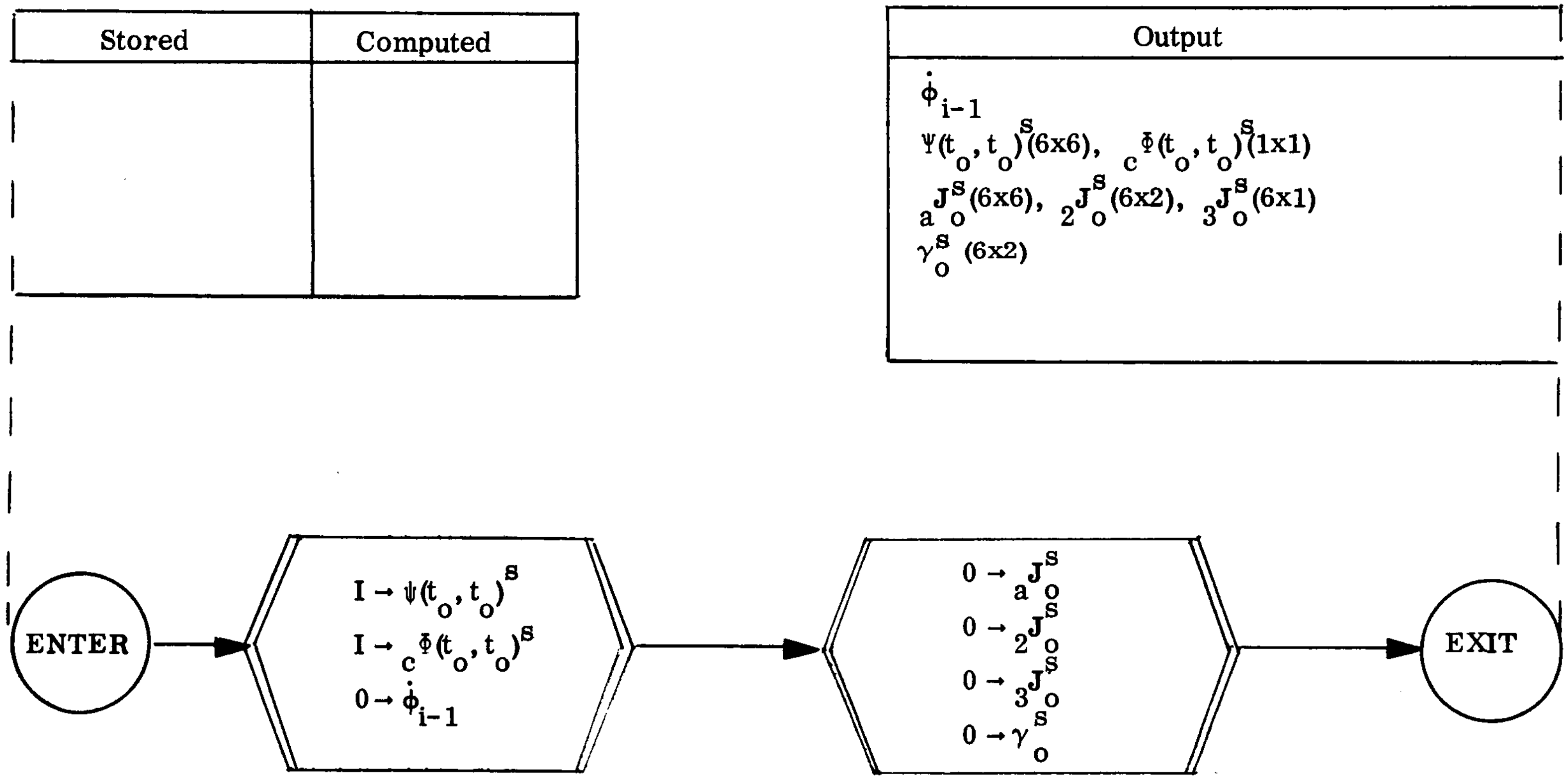
2. $$A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} 0 & \cos\varphi_o & \sin\varphi_o \\ 0 & -\sin\varphi_o & \cos\varphi_o \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} U_{vx} & U_{vy} & U_{vz} \\ U_{px} & U_{py} & U_{pz} \\ U_{ux} & U_{uy} & U_{uz} \end{bmatrix}$$

3. $$A_3 = \begin{bmatrix} \cos\alpha_{10} & -\sin\alpha_{10} & 0 \\ \sin\alpha_{10} & \cos\alpha_{10} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\alpha_{20} & 0 & \sin\alpha_{20} \\ 0 & 1 & 0 \\ -\sin\alpha_{20} & 0 & \cos\alpha_{20} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{30} & -\sin\alpha_{30} \\ 0 & \sin\alpha_{30} & \cos\alpha_{30} \end{bmatrix}$$

4. $A_4 = A_3 A_2$

5. $$\begin{bmatrix} \underline{P}_{Io} \\ \underline{Y}_{Ao} \\ \underline{R}_{Oo} \end{bmatrix} \triangleq A_4 \begin{bmatrix} \underline{i} \\ \underline{j} \\ \underline{k} \end{bmatrix}$$

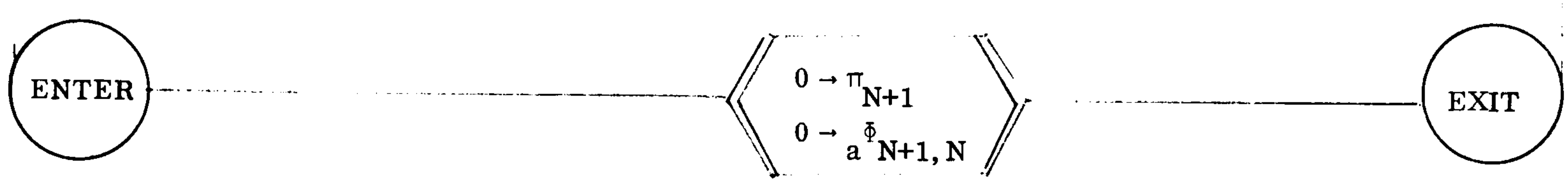
6. $$A_5 = \begin{bmatrix} \sin A_o & \cos A_o & 0 \\ -\cos A_o & \sin A_o & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sin\lambda_o & 0 & -\cos\lambda_o \\ 0 & 1 & 0 \\ \cos\lambda_o & 0 & \sin\lambda_o \end{bmatrix} \begin{bmatrix} \cos\mu_o & \sin\mu_o & 0 \\ -\sin\mu_o & \cos\mu_o & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$



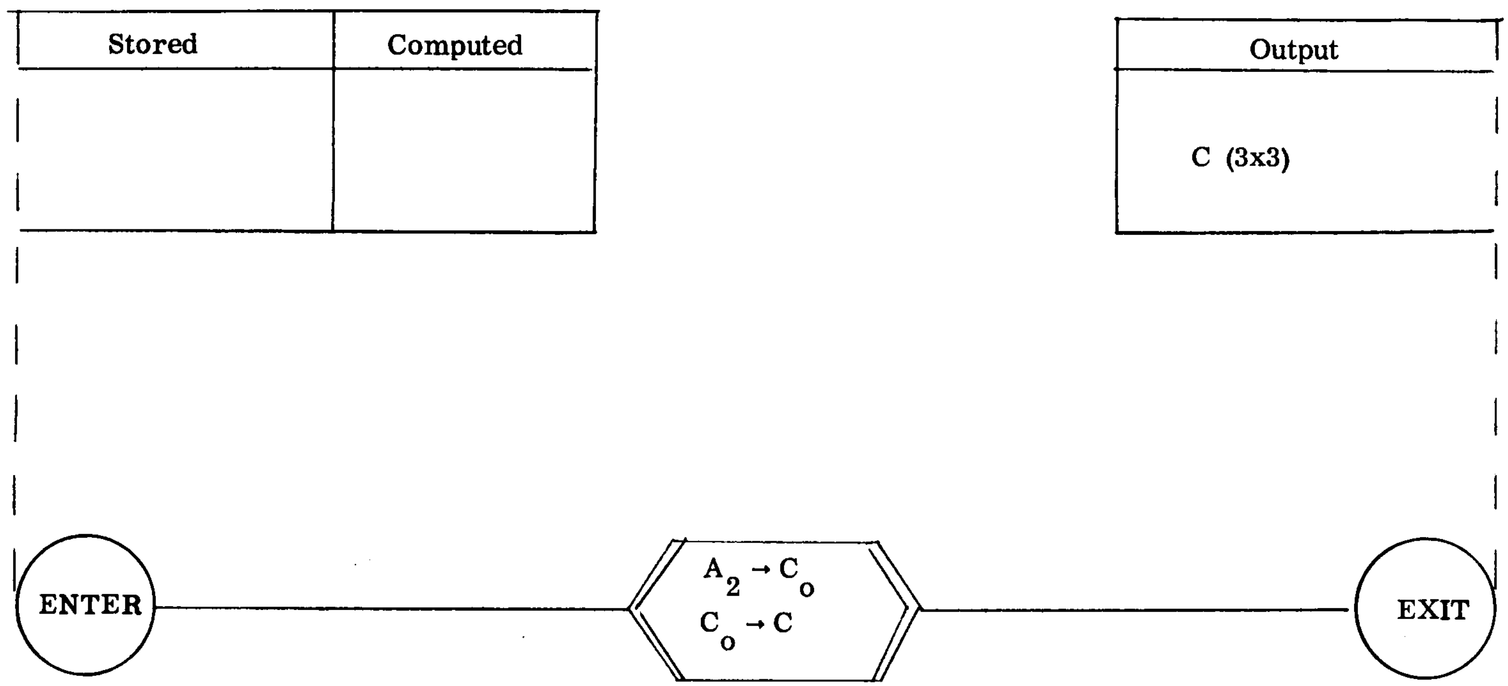
3.3.2.2.2 Linear System Initilization - Block B.2

Stored	Computed

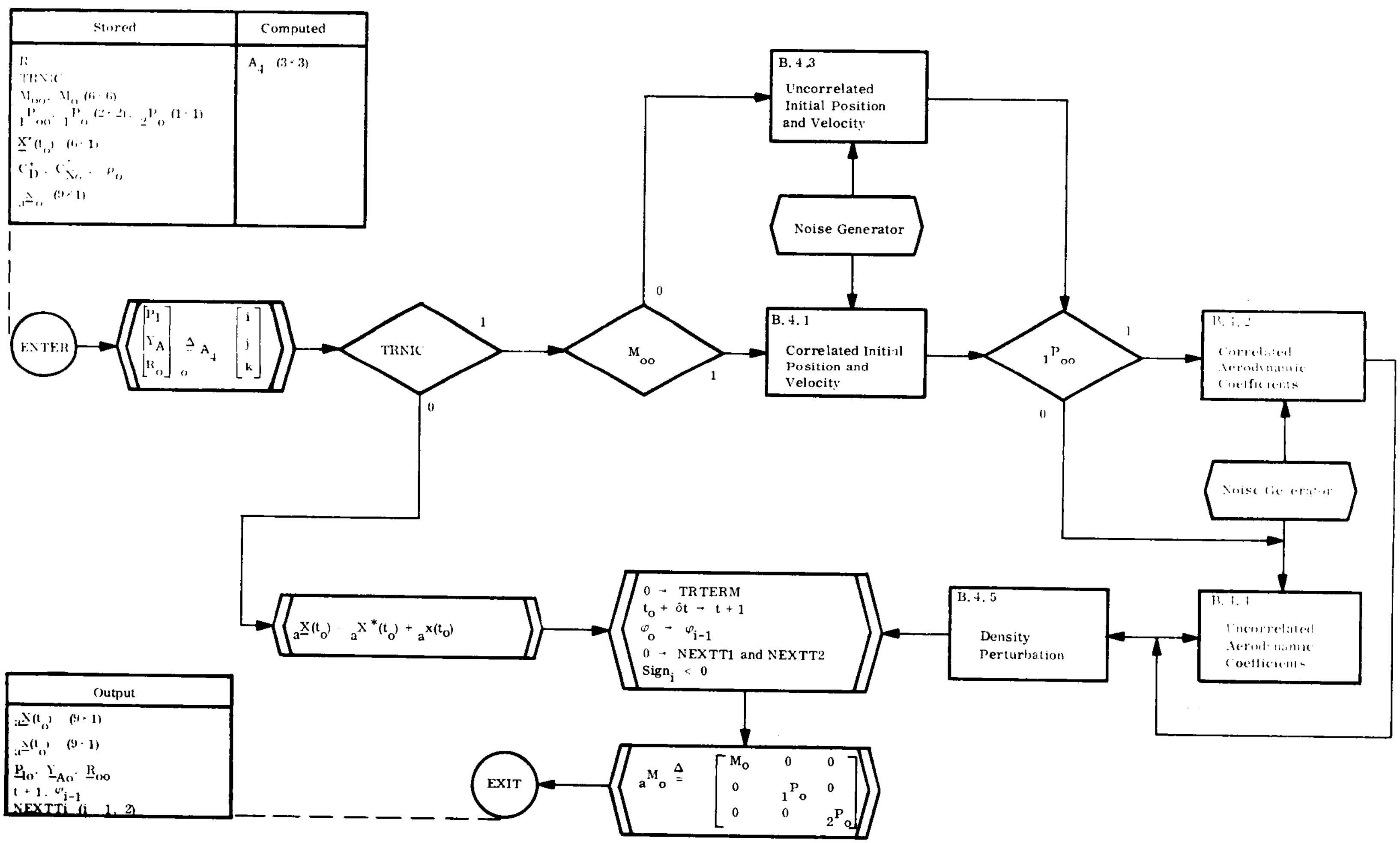
Output
π_{N+1} (9x9) $a^{\Phi}{}_{N+1,N}$ (9x9)



3.3.2.2.3 Guidance Law Initialization - Block B.3



3.3.2.2.4 IMU Error Initialization - Block B.3a



3. 3. 2. 2. 5 Actual Trajectory Initialization - Block B. 4

Block B.4.1 Correlated Initial Position and Velocity

Input: M_o (6x6), $\underline{X}^*(t_o)$ (6x1)

Output: $\underline{x}(t_o)$ (6x1), $\underline{X}(t_o)$ (6x1)

1. M_o is sent to the triangularization subroutine. Output of this routine is a diagonal matrix D_M (6x6) and a lower triangular matrix T_M (6x6).

2. Using the noise generator and the diagonal elements of D_M as variances, generate 6 gaussian random numbers with zero means.

3. Premultiply the vector comprised of the 6 elements by T_M to obtain $\underline{x}(t_o)$.

4. $\underline{X}(t_o) = \underline{X}^*(t_o) + \underline{x}(t_o)$

Block B. 4. 2 Correlated Aerodynamic Coefficients

Input: C^*_{Do}, $C^*_{N\alpha}$, ${}_1P_o(2x2)$

Output: C_{Do}, $C_{N\alpha}$

1. Use the procedures described in Block B. 4. 1 to generated δC_{Do} and $\delta C_{N\alpha}$.

2. $C_D = C^*_{Do} + \delta C_{Do}$

3. $C_{N\alpha} = C^*_{N\alpha} + \delta C_{N\alpha}$

Block B. 4. 3 Uncorrelated Initial Conditions

Input: M_o (6x6), $\underline{X}^*(t_o)$ (6x1)

Output: $\underline{x}(t_o)$ (6x1), $\underline{X}(t_o)$ (6x1)

1. Using the diagonal elements of M_o as the variances and the noise generator, generate 6 gaussian random numbers which form the vector $\underline{x}(t_o)$.

2. $\underline{X}(t_o) = \underline{X}^*(t_o) + \underline{x}(t_o)$

Block B.4.4 Uncorrelated Aerodynamic Coefficients

Input: C^*_{Do}, $C^*_{N\alpha}$, ${}_1P_o$(2x2)

Output: C_{Do}, $C_{N\alpha}$

1. Using the diagonal elements of ${}_1P_o$ as the variances and the noise generator, generate 2 gaussian random numbers of δC_{Do} and $\delta C_{N\alpha}$.

2. $$C_{Do} = C^*_{Do} + \delta C_{Do}$$

$$C_{N\alpha} = C^*_{N\alpha} + \delta C_{N\alpha}$$

Block B.4.5 Atmospheric Density Perturbation

Input: ${}_2P_o$ (1x1)

Output: $\delta\rho_o$

1. Using the noise generation and ${}_2P_o$ as the variance, generate a gaussian number $\delta\rho_o$.

3.3.2.2.7 Electromagnetic Sensors - Block B.5

No initialization required for this block.

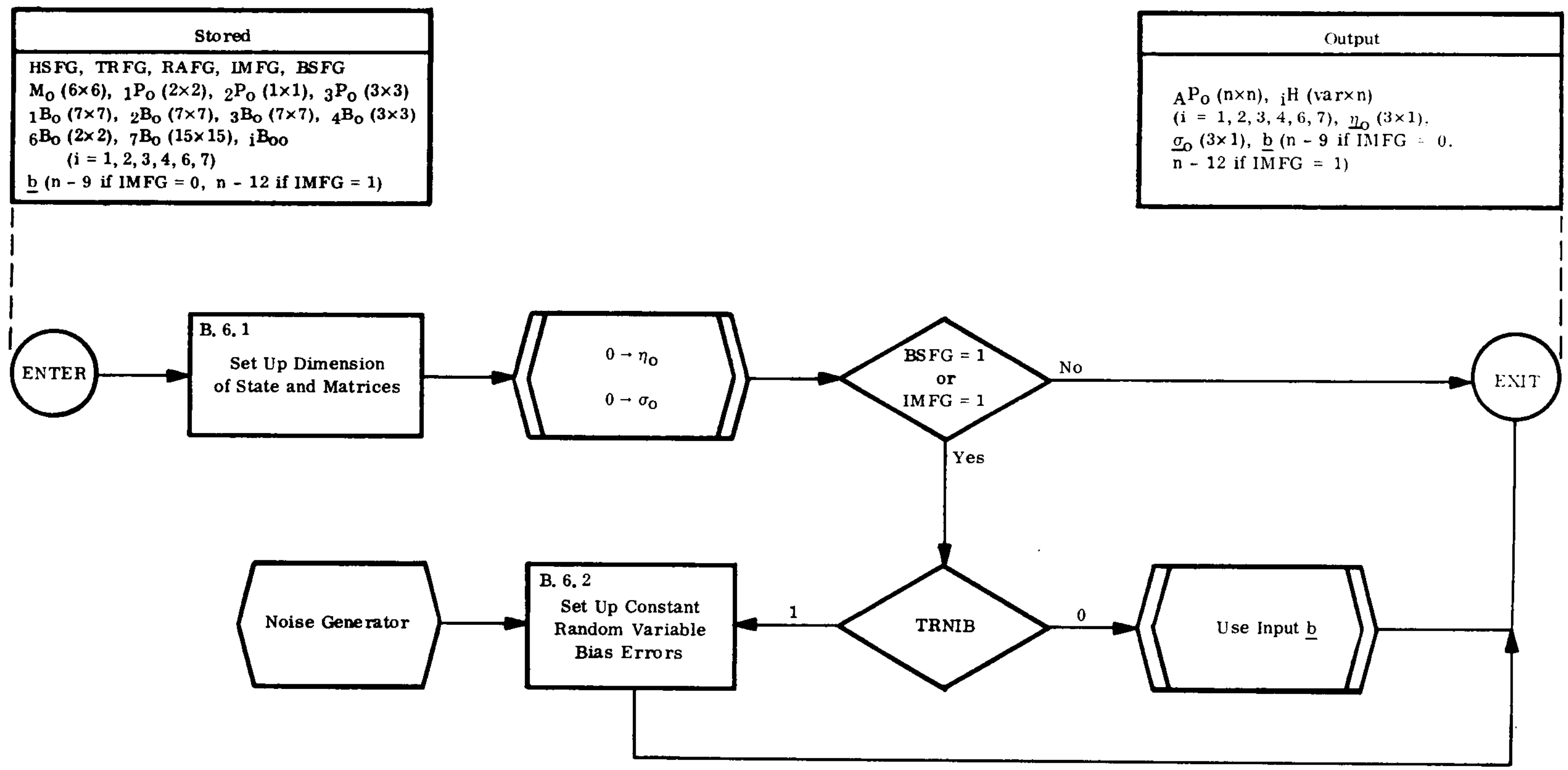


Figure 3. 3. 2. 2. 7 Navigation Initialization - Block B. 6

Block B. 6. 1 Navigation Initialization

Input: HSFG, TRFG, RAFG, IMFG, BSFG, $M_o(6x6)$, ${}_1P_o(2x2)$, ${}_2P_o(1x1)$, ${}_3P_o(3x3)$, ${}_1B_o(7x7)$, ${}_2B_o(7x7)$, ${}_3B_o(7x7)$, ${}_4B_o(3x3)$, ${}_6B_o(2x2)$, ${}_7B_o(15x15)$

Output: ${}_AP_o(nxn)$, ${}_iH$ (var x n)(i=1, 2, 3, 4, 6, 7), $\underline{\eta}_o(3x1)$, $\underline{\sigma}_o(3x1)$, $\underline{b}$ (m-9 if IMFG = 0, n-12 if IMFG = 1)

The overall dimension of the state vector and matrices and vectors shall be established in this block by means of the bias flag and the flags used to call any of the 6 aiding instruments. Throughout this document the instruments are referred to by the number assigned to it below.

i=1	ground tracking system No. 1
i=2	ground tracking system No. 2
i=3	ground tracking system No. 3
i=4	horizon sensor
i=6	radio altimeter
i=7	IMU (inertial measurement unit)

The dimension, n, of the state vector varies from a minimum of 9, when there are no instrument or IMU bias errors to a maximum of 34. This restriction of the maximum dimension means that not all aiding instruments, including the IMU, can be used in one computer run if bias errors are simulated since this would make the dimension of the state equal to 53. The following vectors and matrices have at least one dimension defined by n.

1. ${}_A\underline{x}(t_k)$, ${}_A\hat{\underline{x}}(t_k)$ and all other state related vectors are (nx1).
2. ${}_AP(t_k)$, ${}_AP'(t_k)$, ${}_AM(t_k)$ are (nxn).
3. ${}_iH(t_k)$ is (m_i x n) (i=1, 2, 3, 4, 6, 7)
4. ${}_iK(t_k)$ is (n x m_i) (i=1, 2, 3, 4, 6, 7)

where the m_i has a magnitude equal to the measurement made by the i^{th} aiding instrument.

i = 1, 2, 3	→ TRFG = 1, 2, 3	$m_i = 4$
i = 4	→ HSFG = 1	$m_4 = 3$

$i = 6 \quad \rightarrow \text{RAFG} = 1 \qquad m_6 = 2$

$i = 7 \quad \rightarrow \text{IMFG} = 1 \qquad m_7 = 3$

Table B. 6 indicates the form of the i^{th} observation matrix ${}_iH$ (i=1, 2, 3, 4, 6, 7). Each observation matrix has n columns where n is defined using the following rules.

The $\underline{x}_k$, $\underline{b}'$, $\underline{c}_k$ columns are always used.

If BSFG = 1, the ${}_i\underline{d}$ and ${}_i\underline{\alpha}$ columns corresponding to the aiding instrument requested by the instrument flags are used.

If BSFG = 0, the ${}_i\underline{d}$ and ${}_i\underline{\alpha}$ columns are not used.

The $\underline{\epsilon}$ and $\underline{\eta}$ columns are used if IMFG = 1 regardless of BSFG.

The flags also define the structure of the covariance matrix at time $t = t_o$, ${}_AP_o$. This is an input quantity and may be partitioned in the following form

$$
1. \quad {}_AP_o = \begin{bmatrix}
M_o(6\text{x}6)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_1P_o(2\text{x}2)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_2P_o(1\text{x}1)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_3P_o(3\text{x}3) = \text{all zeros}\ 0 \longrightarrow \\
\longleftarrow 0\ {}_1B_o(7\text{x}7)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_2B_o(7\text{x}7)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_3B_o(7\text{x}3)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_4B_o(3\text{x}3)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_6B_o(2\text{x}2)\ 0 \longrightarrow \\
\longleftarrow 0\ {}_7B_o(15\text{ x }15)\ 0 \longrightarrow
\end{bmatrix}
$$

Because the dimension of the state is restricted to 34, not all of the above submatrices can be used simultaneously. When a matrix is defined, however, it has the relative location indicated in (1) above with respect to other matrices which are present. Four rules analogous to th se defining the n dimension of the observation matrices are given below.

1. M_o, ${}_1P_o$, ${}_2P_o$ are always used corresponding to a minimum dimension of 9.
2. If BSFG = 1, the ${}_iB_o$ corresponding to the instrument flags are used.

$i = 1, 2, 3 \rightarrow \text{TRFG} = 1, 2, 3$ respectively

$i = 4 \quad \rightarrow \text{HSFG} = 1$

$i = 6 \rightarrow$ RAFG = 1

$i = 7 \rightarrow$ IMFG = 1

3. If BSFG = 0, the ${}_{i}B_{o}$ (i=1, 2, 3, 4, 6) are not used.
4. If IMFG = 1, ${}_{3}P_{o}$ and ${}_{7}B_{o}$ are used regardless of the BSFG value.

Table B. 6 showing the construction of the observation matrices is shown on the next page.

Item		state variable	drag coefficients	density	central system noise	tracker 1 location + bias errors	tracker 2 location + bias errors	tracker 3 location + bias errors	horizon sensor bias	radio altimeter bias	IMU bias
		$\underline{x}_k$	$\underline{b}'$	$\underline{c}_k$	$\underline{\eta}_k$	$_1\underline{d}$	$_2\underline{d}$	$_3\underline{d}$	$_4\underline{\alpha}$	$_6\underline{\alpha}$	$_7\underline{\epsilon}$
Tracker 1	$_1H$	$_1H_{T1}$ 4×6	0 4×2	0 4×1	0 4×3	$_1H_{T2}$ 4×7	0 4×7	0 4×7	0 4×3	0 4×2	0 4×15
Tracker 2	$_2H$	$_2H_{T1}$ 4×6	0 4×2	0 4×1	0 4×3	0 4×7	$_2H_{T2}$ 4×7	0 4×7	0 4×3	0 4×2	0 4×15
Tracker 3	$_3H$	$_3H_{T1}$ 4×6	0 4×2	0 4×1	0 4×3	0 4×7	0 4×7	$_3H_{T2}$ 4×7	0 4×3	0 4×2	0 4×15
Horizon Sensor	$_4H$	H^H 3×6	0 3×2	0 3×1	0 3×3	0 3×7	0 3×7	0 3×7	$HB^{H}=(I)$ 3×3	0 3×2	0 3×15
Radio Altimeter	$_6H$	R^H 2×6	0 2×2	0 2×1	0 2×3	0 2×7	0 2×7	0 2×7	0 2×3	$RB^{H}=(I)$ 2×2	0 2×15
IMU	$_7H$	$_aJ_k$ 3×6	$_2J_k$ 3×2	$_3J_k$ 3×1	I 3×3	0 3×7	0 3×7	0 3×7	0 3×3	0 3×2	$\int_{t_o}^{t_k} G dt$ 3×15

Table B. 6. Setup of DIMFG and Observation Matrix Form

Block B. 6. 2 - Set up Constant Random Variable Bias Errors

Input: ${}_iB_{oo}$, ${}_iB_o$ (i=1, 2, 3, 4, 6, 7)

Output: $\underline{b}$ ([n-9]) x 1 if IMFG = 0 or ([n-12] x 1 if IMFG = 1)

When ${}_iB_o$ is set up in Block B. 6. 1, i.e., when BSFG = 1 and/or IMFG = 1 the appropriate bias vector, $\underline{b}$, must be generated.

$$\underline{b} = \begin{bmatrix} {}_1\underline{d} \\ {}_2\underline{d} \\ {}_3\underline{d} \\ {}_4\underline{\alpha} \\ {}_6\underline{\alpha} \\ {}_7\underline{\epsilon} \end{bmatrix} \begin{matrix} (7x1) \\ (7x1) \\ (7x1) \\ (3x1) \\ (2x1) \\ (15x1) \end{matrix}$$

The components of each subvector are generated from the gaussian noise generator using the statistics ${}_iB_o$. When ${}_iB$ is nondiagonal (${}_iB_{oo}$ = 1), the matrix must be factored into a diagonal matrix and a lower triangular matrix as indicated in Block B. 4. If the matrix is diagonal (${}_iB_{oo}$ = 0), the diagonal elements are variances of the components.

3.3.2.2.8 Initialization for Guidance - Block B.7

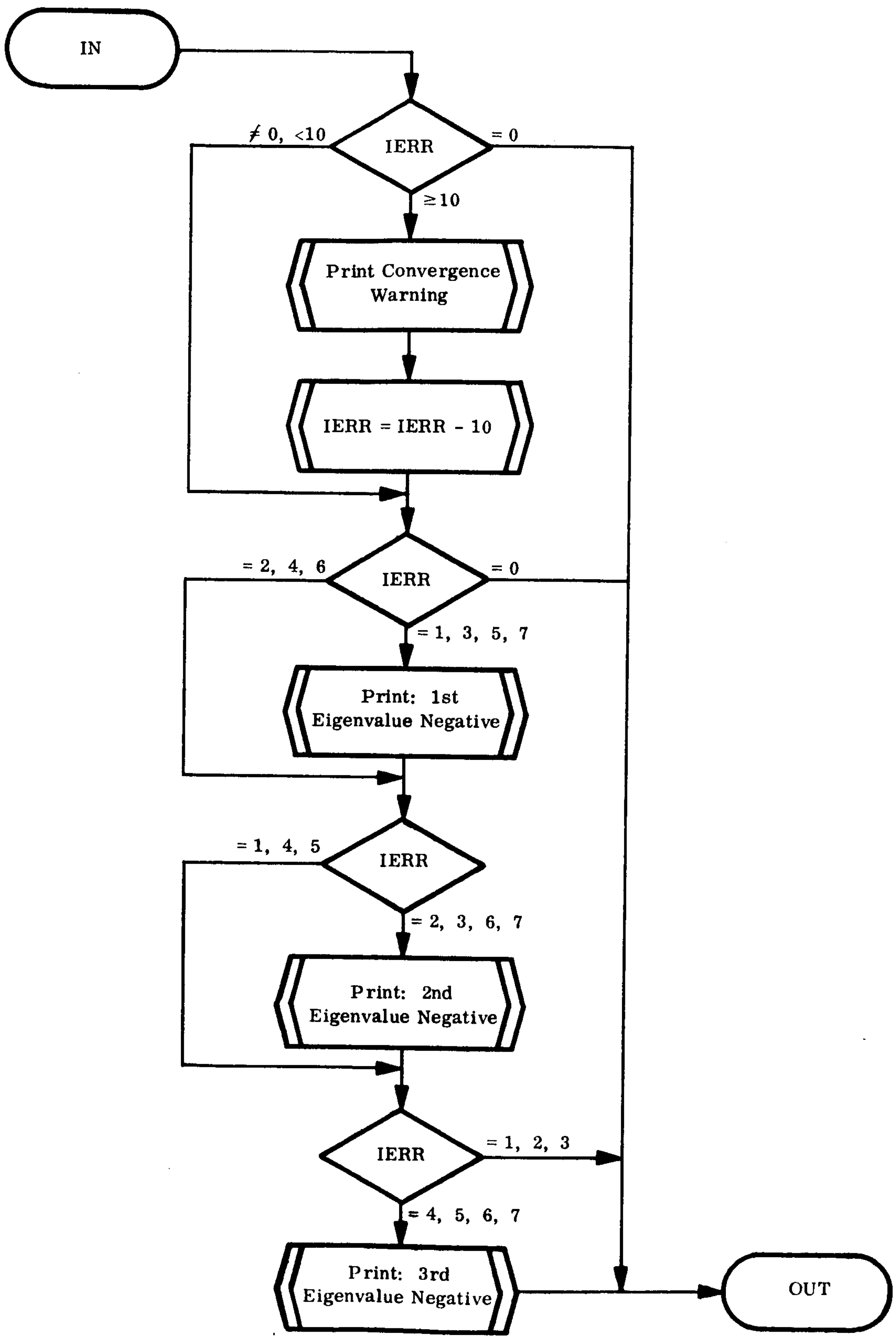
No initialization is required.

3.3.3 Output - Block C

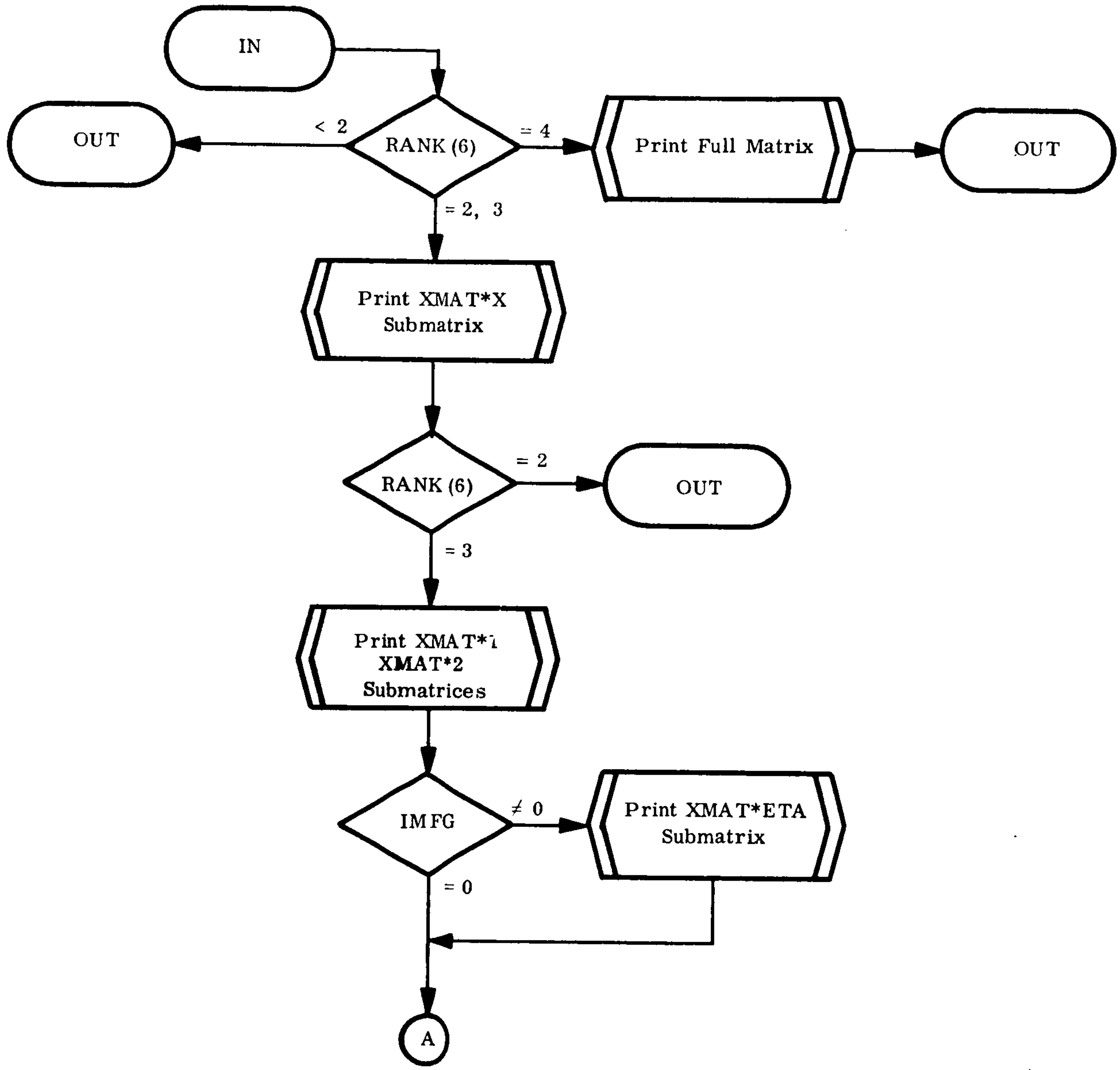
The output of this program consists of the data which is stored on the output tape, tape 4, and is printed by the tape edit routine. During some operational modes of the program, i.e., when generating only a new nominal and/or guidance law or IMU error matrices, the only output is the supplementary data which is stored on the output tape. Certain data is printed "on-line" for the convenience of the user so that he may observe a subset of the output during the operation of the program. This data consists of position, velocity and trajectory constraint data which can be used by the engineer to follow the progress of the program during operation.

Each block has a rank number assigned to the output from that block which specifies how much data is desired for print from the tape edit routine. It is possible, therefore, to print a small amount of data for a run and save the output tape for further, more extensive tape editing if this should prove desirable. Rank number assignment is described in detail in paragraph 4.9.

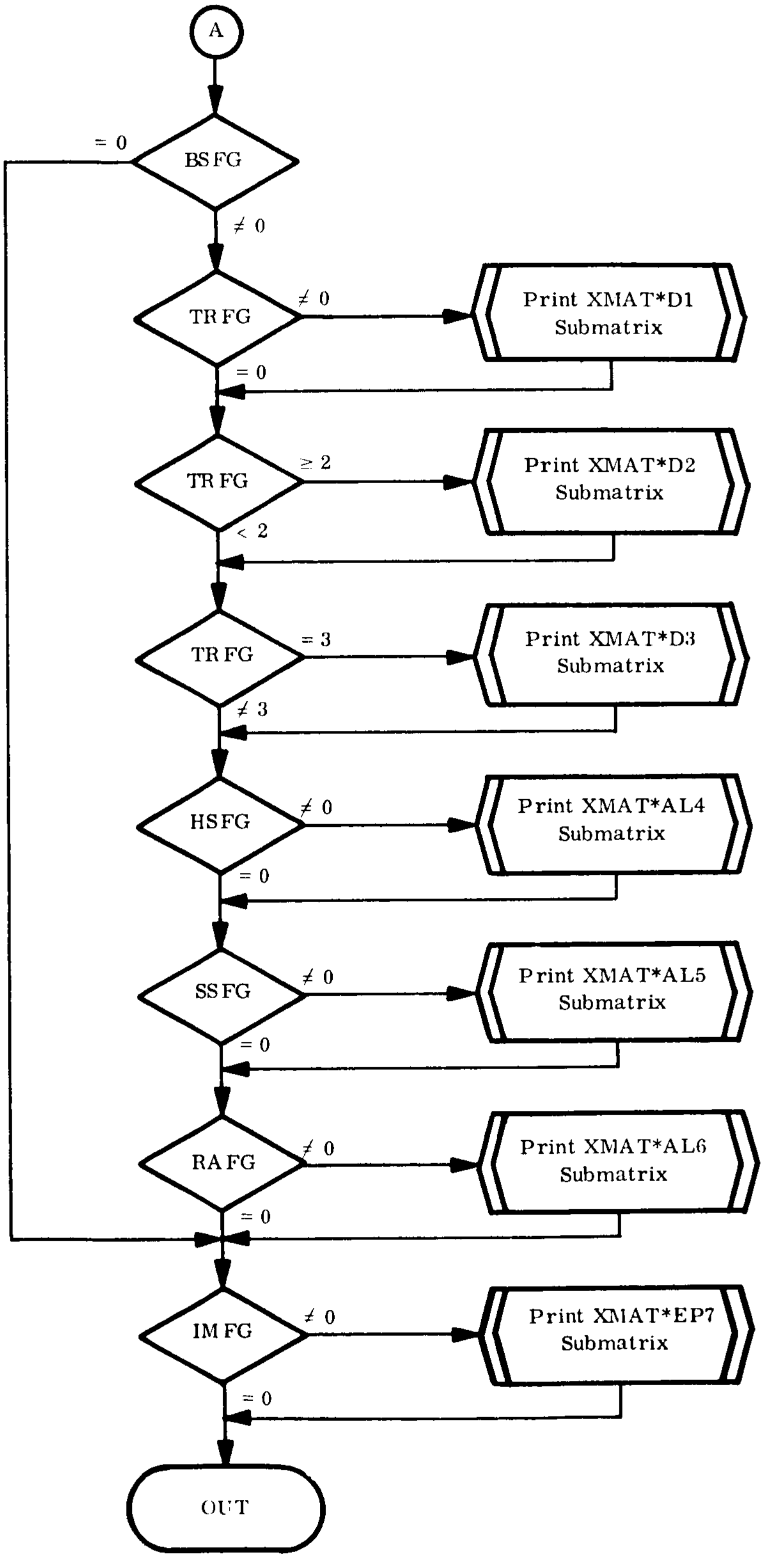
The tape edit routine is described in detail by means of flow charts in the following pages. The format of these flow charts is somewhat different from that of paragraph 3.3.2 and 3.4 because they are slanted primarily to the needs of a programmer. A summary of the function of each of the subroutines is presented in paragraph 5.2.


IN
IERR
≠ 0, <10
= 0
≥ 10
Print Convergence Warning
IERR = IERR - 10
IERR
= 2, 4, 6
= 0
= 1, 3, 5, 7
Print: 1st Eigenvalue Negative
IERR
= 1, 4, 5
= 2, 3, 6, 7
Print: 2nd Eigenvalue Negative
IERR
= 1, 2, 3
= 4, 5, 6, 7
Print: 3rd Eigenvalue Negative
OUT

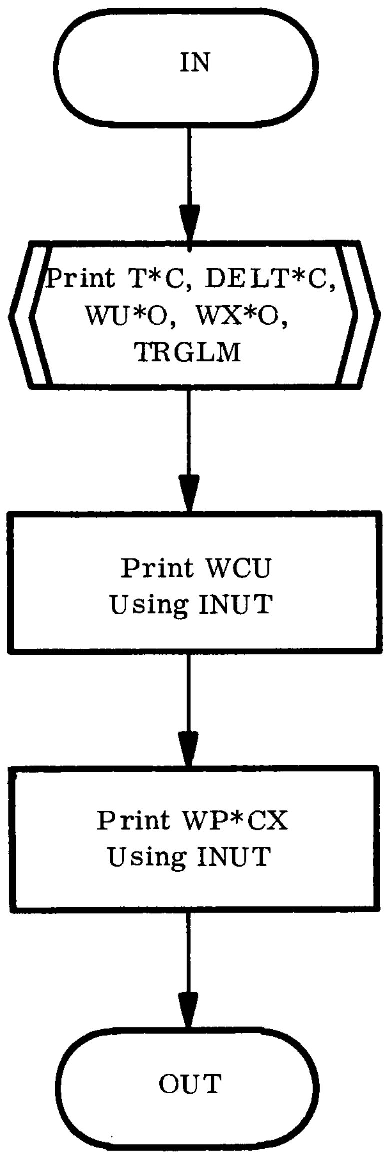
EIGVER


IN
OUT
< 2
RANK (6)
= 4
Print Full Matrix
OUT
= 2, 3
Print XMAT*X
Submatrix
RANK (6)
= 2
OUT
= 3
Print XMAT*1
XMAT*2
Submatrices
IMFG
≠ 0
Print XMAT*ETA
Submatrix
= 0
A

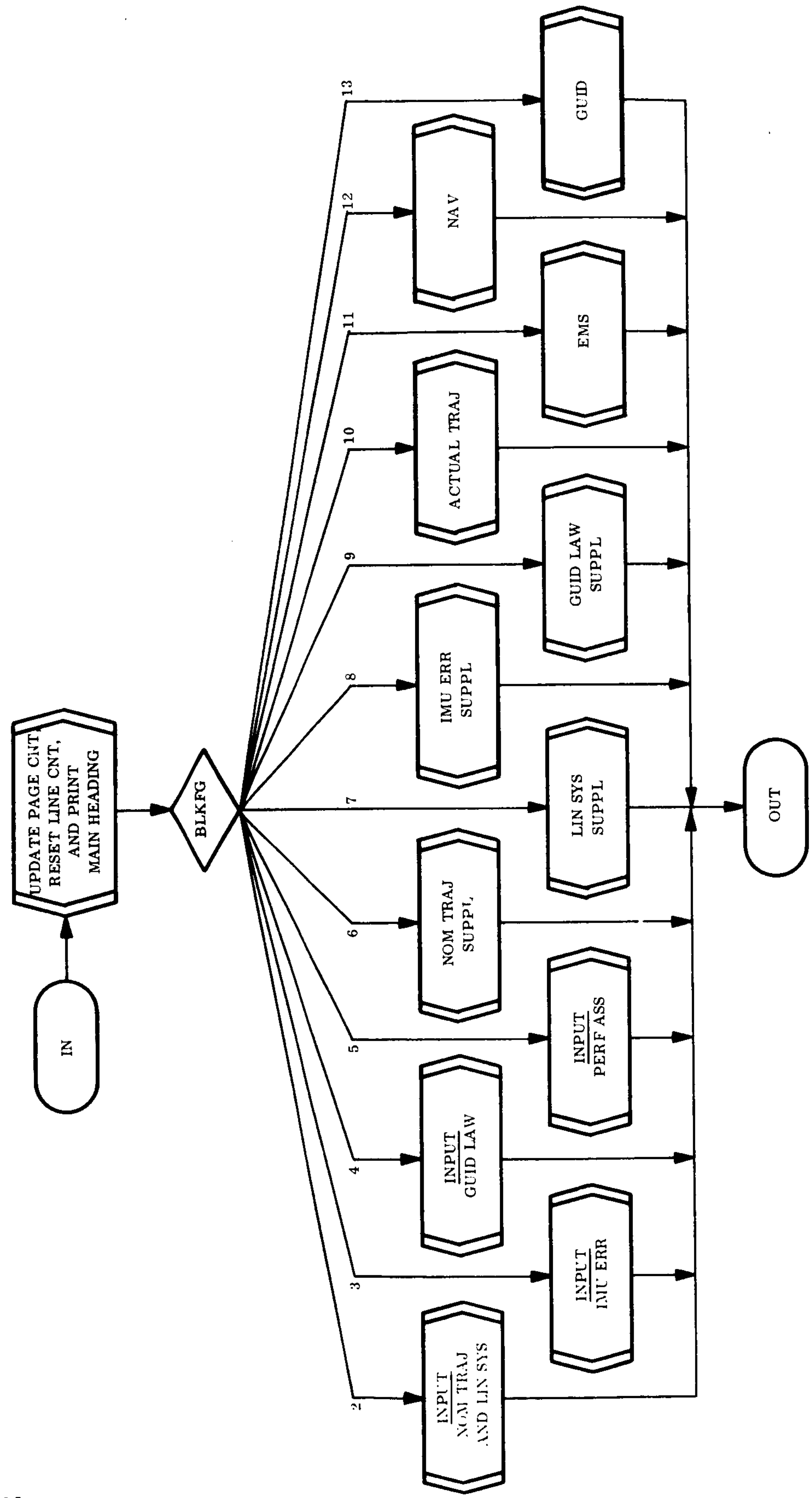
EXERP


A
BS FG
= 0
≠ 0
TR FG
≠ 0
Print XMAT*D1 Submatrix
= 0
TR FG
≥ 2
Print XMAT*D2 Submatrix
< 2
TR FG
= 3
Print XMAT*D3 Submatrix
≠ 3
HS FG
≠ 0
Print XMAT*AL4 Submatrix
= 0
SS FG
≠ 0
Print XMAT*AL5 Submatrix
= 0
RA FG
≠ 0
Print XMAT*AL6 Submatrix
= 0
IM FG
≠ 0
Print XMAT*EP7 Submatrix
= 0
OUT

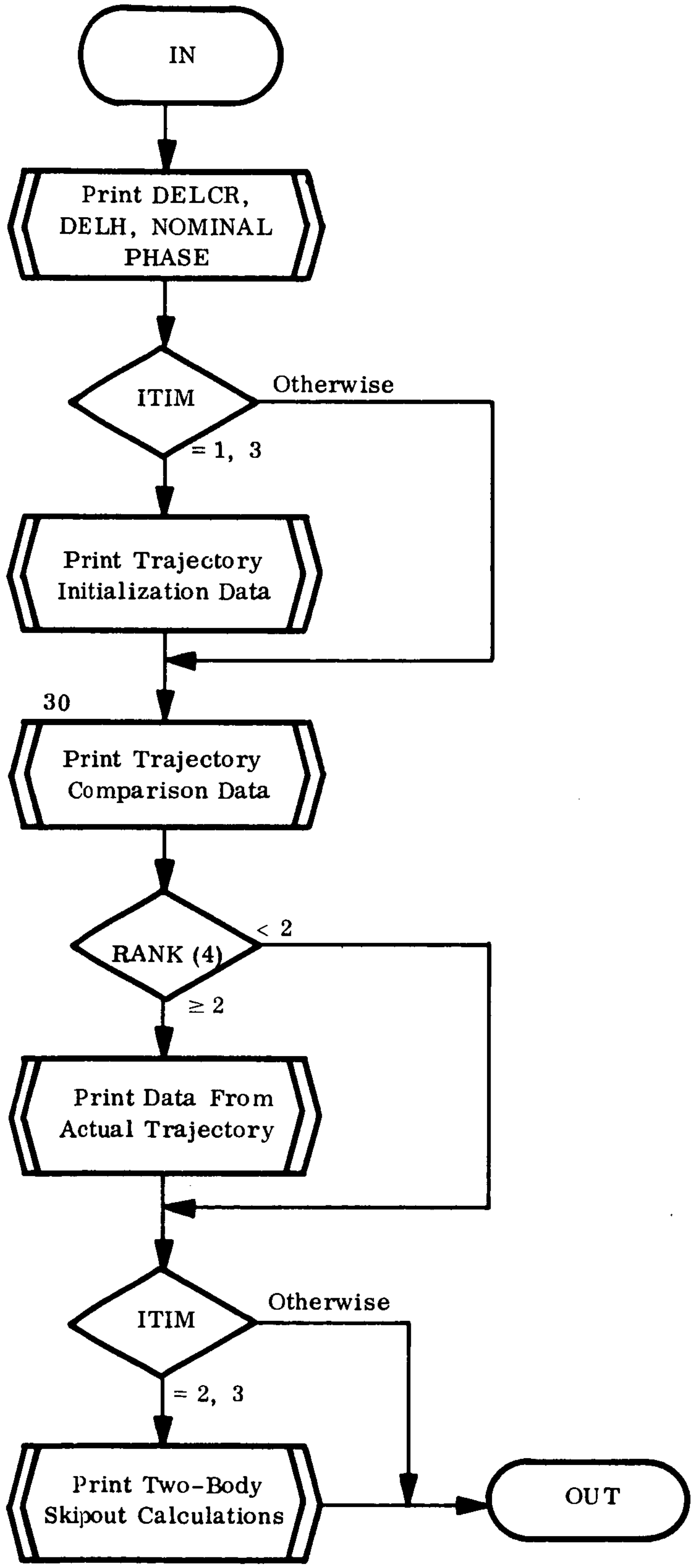
EXERP ~ 2


IN
Print T*C, DELT*C,
WU*O, WX*O,
TRGLM
Print WCU
Using INUT
Print WP*CX
Using INUT
OUT

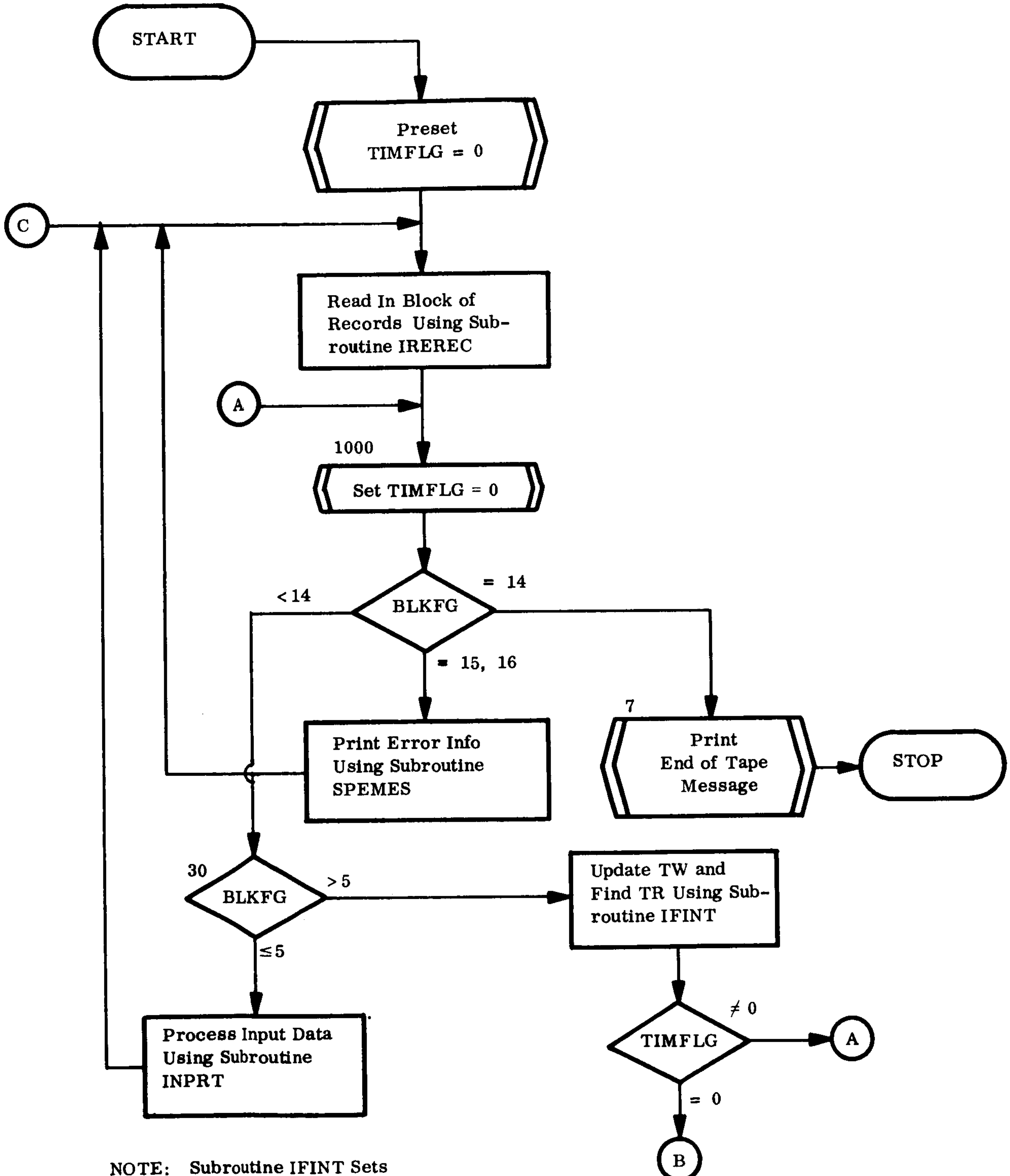
GLAIN


IN
UPDATE PAGE CNT RESET LINE CNT, AND PRINT MAIN HEADING
BLKFG
2
INPUT NOM TRAJ AND LIN SYS
3
INPUT IMU ERR
4
INPUT GUID LAW
5
INPUT PERF ASS
6
NOM TRAJ SUPPL
7
LIN SYS SUPPL
8
IMU ERR SUPPL
9
GUID LAW SUPPL
10
ACTUAL TRAJ
11
EMS
12
NAV
13
GUID
OUT

HEADL

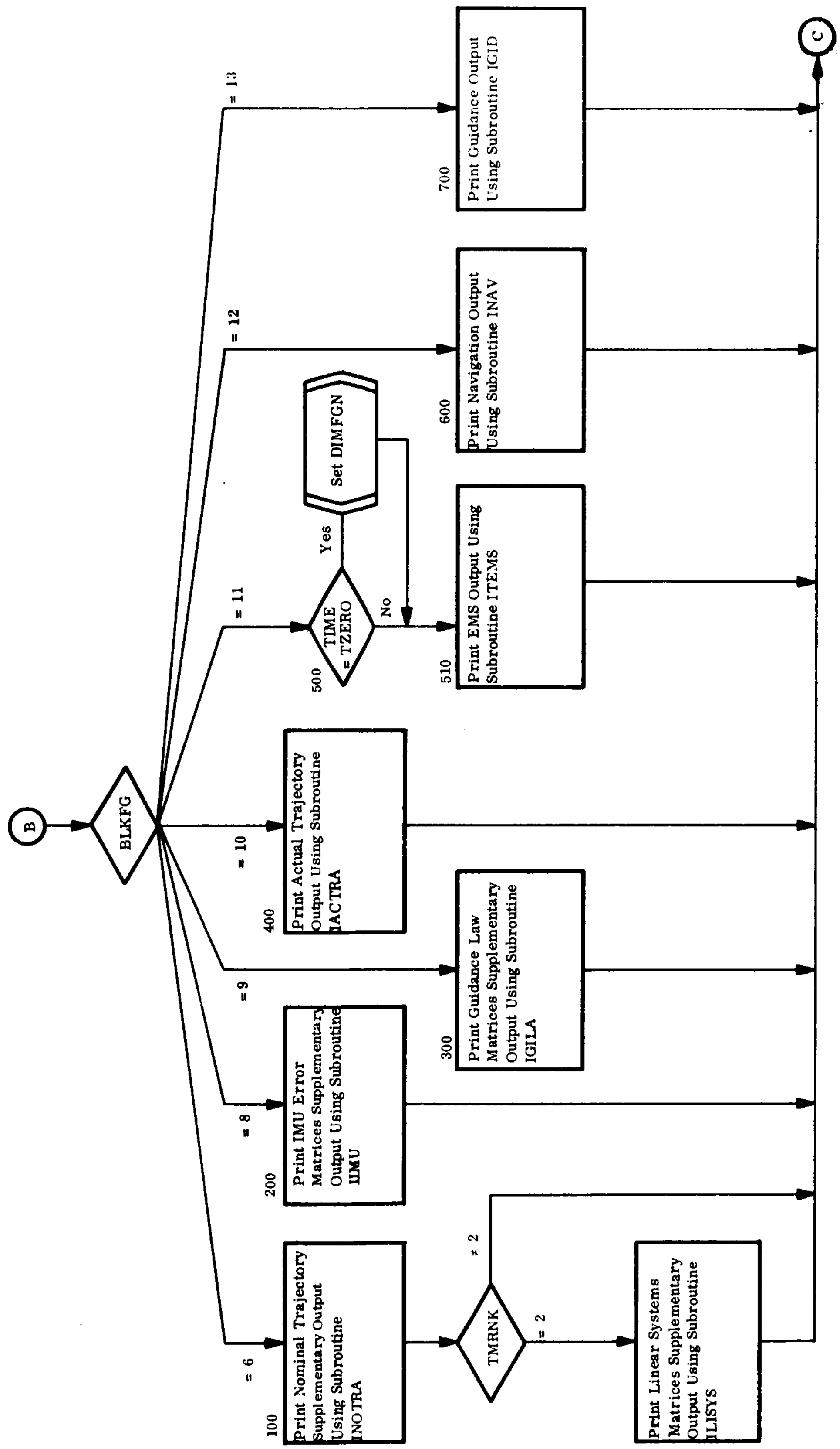

IN
Print DELCR, DELH, NOMINAL PHASE
ITIM
Otherwise
= 1, 3
Print Trajectory Initialization Data
30
Print Trajectory Comparison Data
RANK (4)
< 2
≥ 2
Print Data From Actual Trajectory
ITIM
Otherwise
= 2, 3
Print Two-Body Skipout Calculations
OUT

IACTRA

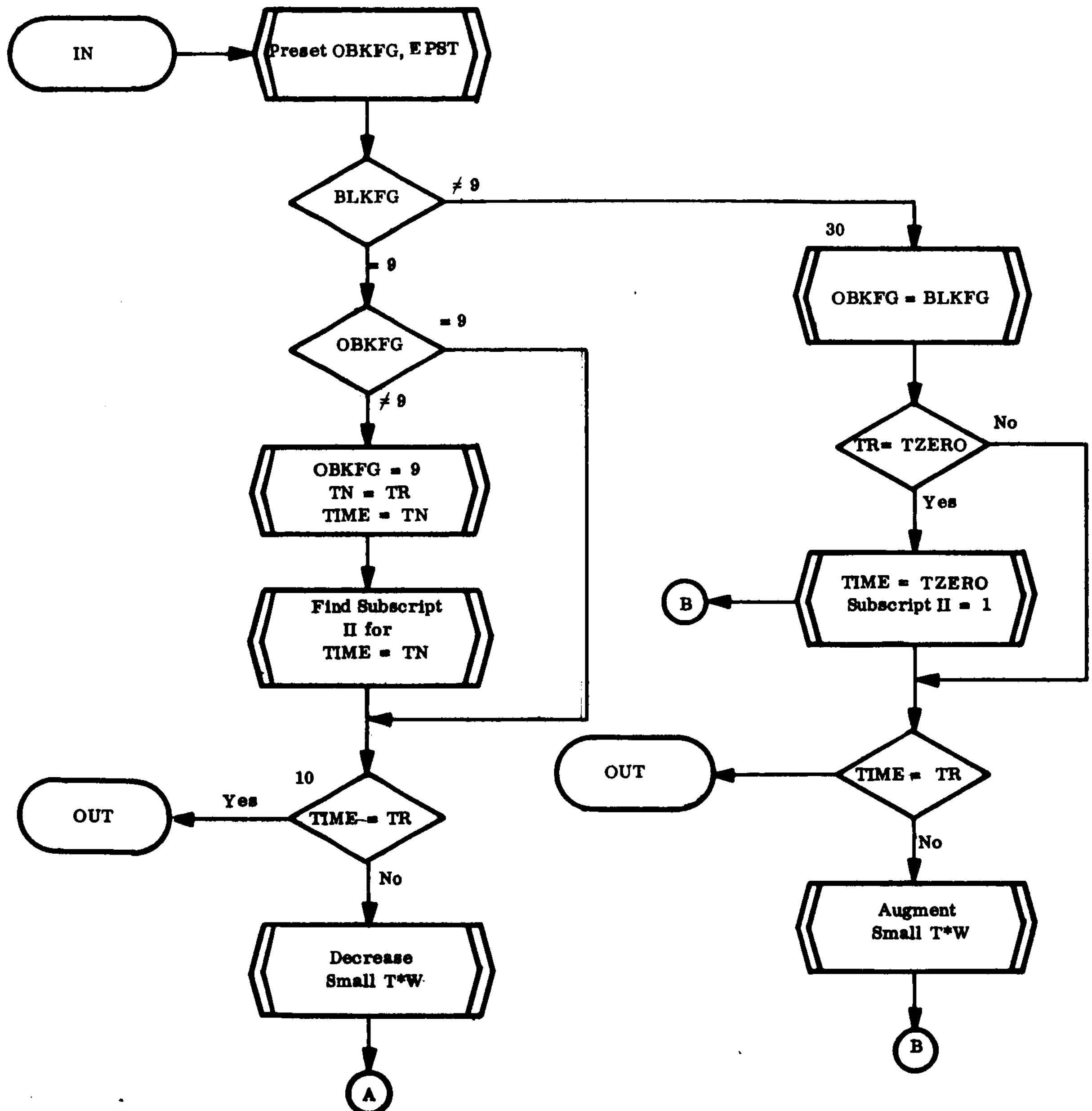


NOTE: Subroutine IFINT Sets TIMFLG ≠ 0 if the Next Block of Records Has Already Been Read-In

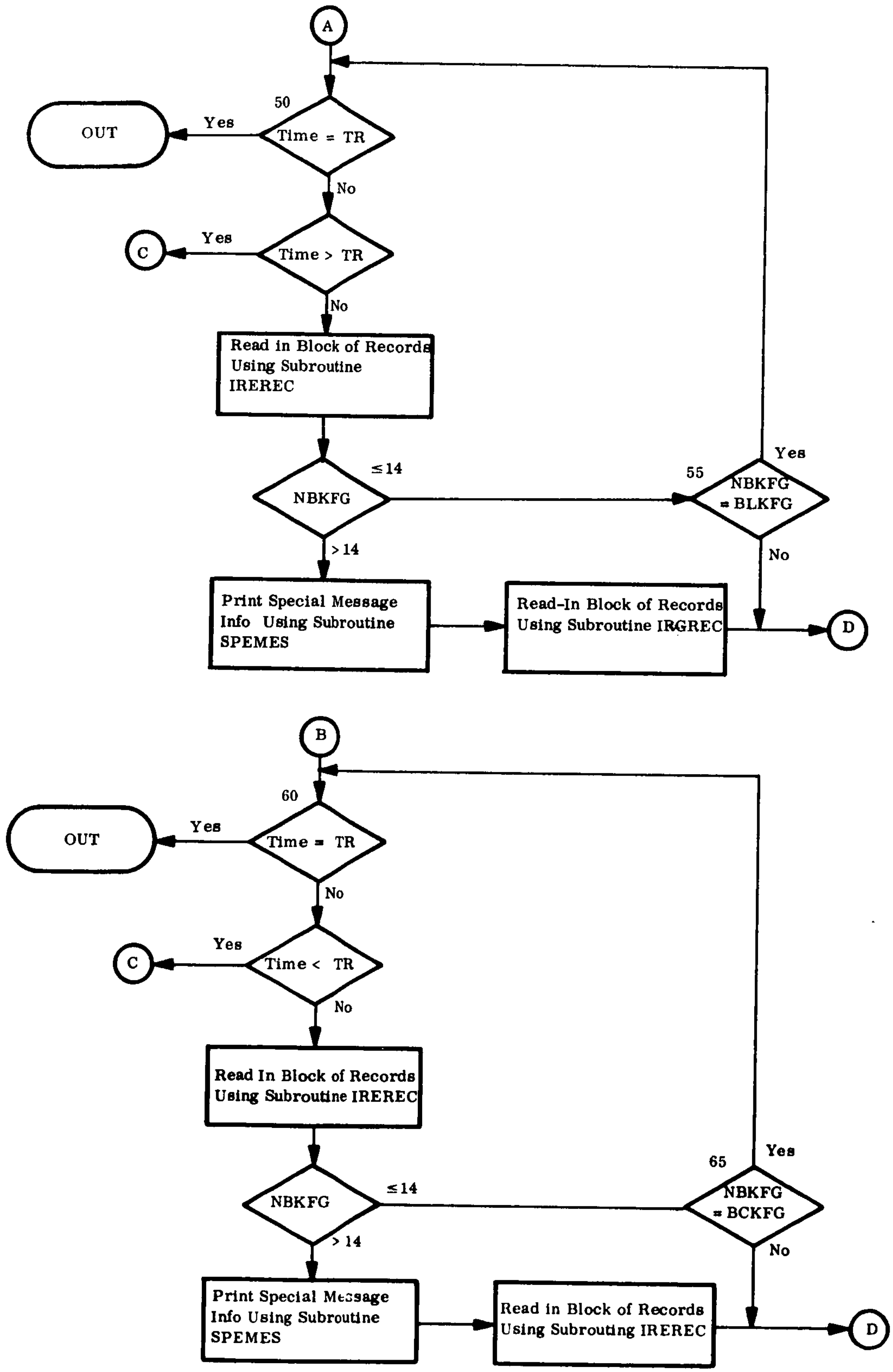
IEDT


B
BLKFG
= 6
= 8
= 9
= 10
= 11
= 12
= 13
100
Print Nominal Trajectory Supplementary Output Using Subroutine INOTRA
200
Print IMU Error Matrices Supplementary Output Using Subroutine IIMU
300
Print Guidance Law Matrices Supplementary Output Using Subroutine IGILA
400
Print Actual Trajectory Output Using Subroutine IACTRA
500
TIME = TZERO
Yes
No
Set DIMFGN
510
Print EMS Output Using Subroutine ITEMS
600
Print Navigation Output Using Subroutine INAV
700
Print Guidance Output Using Subroutine IGID
TMRNK
≠ 2
= 2
Print Linear Systems Matrices Supplementary Output Using Subroutine ILISYS
C

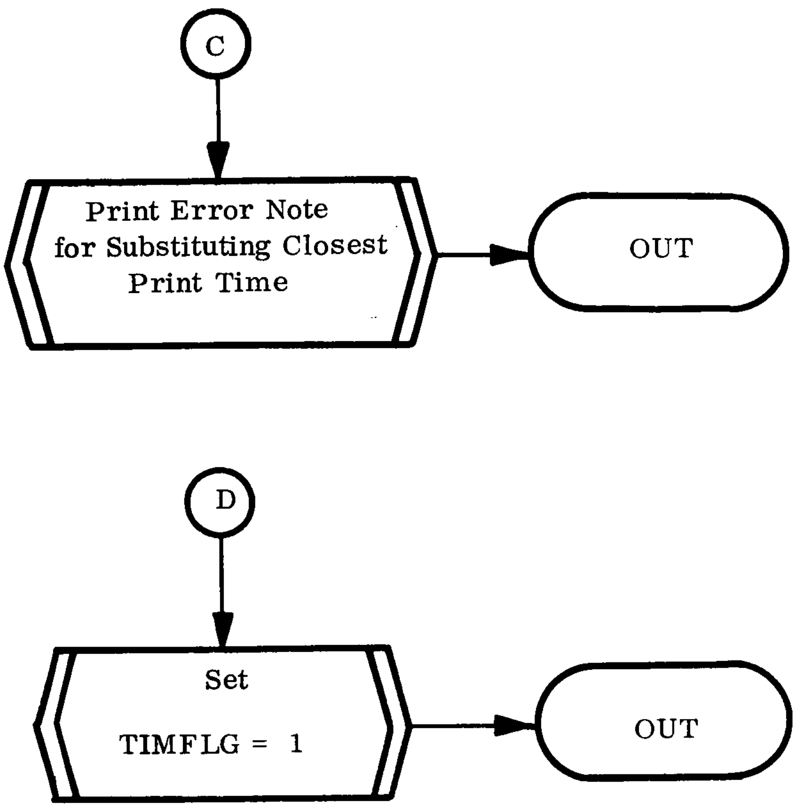
IEDT ~ 2


IN
Preset OBKFG, EPST
BLKFG
≠ 9
= 9
30
OBKFG = BLKFG
OBKFG
= 9
≠ 9
OBKFG = 9
TN = TR
TIME = TN
TR= TZERO
No
Yes
TIME = TZERO
Subscript II = 1
B
Find Subscript
II for
TIME = TN
OUT
TIME = TR
10
OUT
Yes
TIME = TR
No
No
Augment
Small T*W
Decrease
Small T*W
B
A

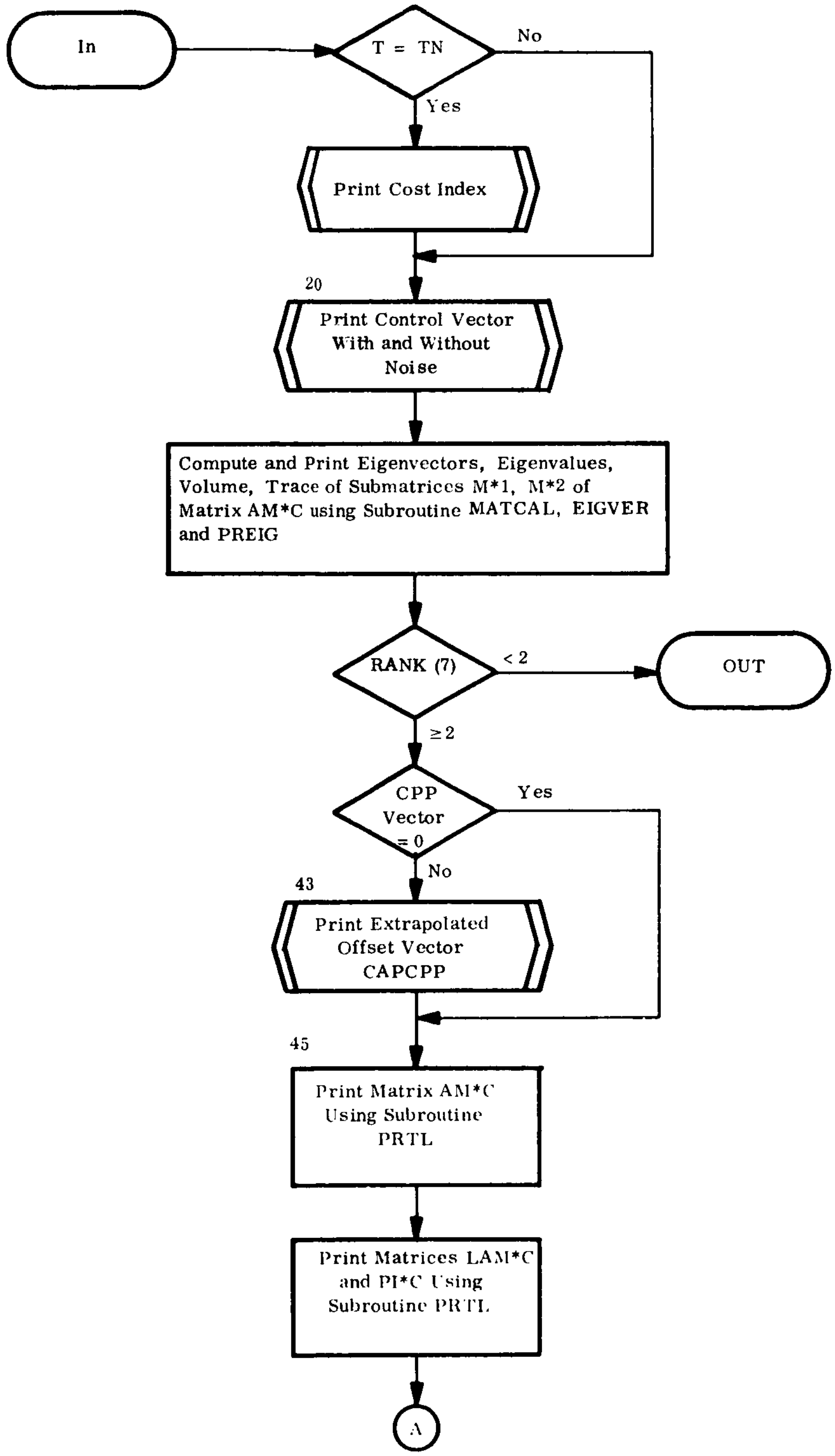
IFINT


A
50
OUT
Yes
Time = TR
No
C
Yes
Time > TR
No
Read in Block of Records Using Subroutine IREREC
NBKFG
≤14
>14
55
Yes
NBKFG = BLKFG
No
Print Special Message Info Using Subroutine SPEMES
Read-In Block of Records Using Subroutine IRGREC
D
B
60
OUT
Yes
Time = TR
No
C
Yes
Time < TR
No
Read In Block of Records Using Subroutine IREREC
NBKFG
≤14
> 14
65
Yes
NBKFG = BCKFG
No
Print Special Message Info Using Subroutine SPEMES
Read in Block of Records Using Subrouting IREREC
D

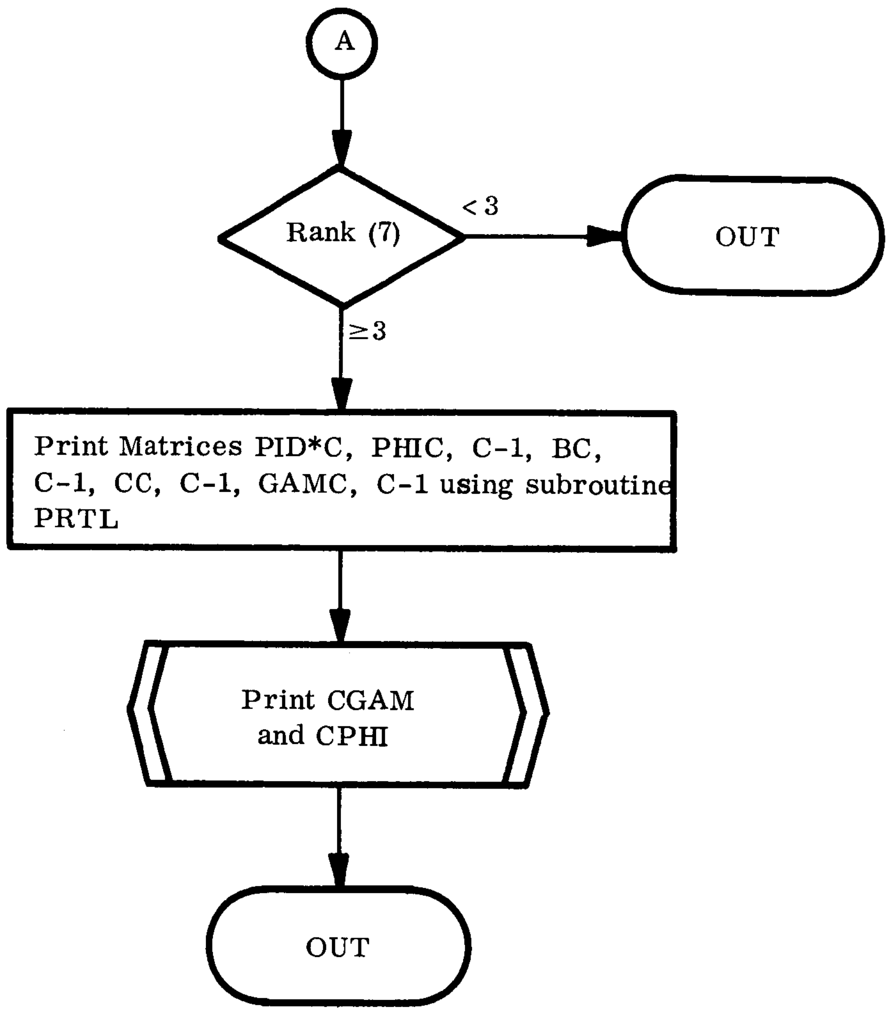
IFINT ~ 2


C
Print Error Note for Substituting Closest Print Time
OUT
D
Set
TIMFLG = 1
OUT

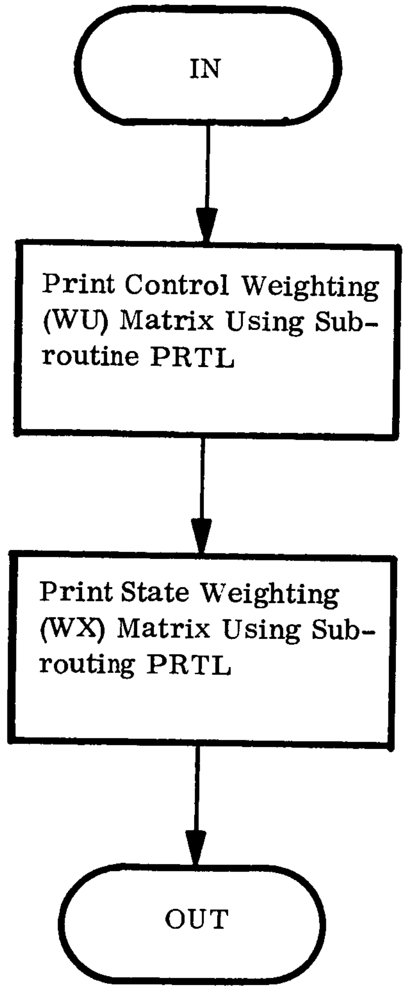
IFINT ~ 3


In
T = TN
No
Yes
Print Cost Index
20
Print Control Vector With and Without Noise
Compute and Print Eigenvectors, Eigenvalues, Volume, Trace of Submatrices M*1, M*2 of Matrix AM*C using Subroutine MATCAL, EIGVER and PREIG
RANK (7)
< 2
OUT
≥ 2
CPP Vector = 0
Yes
No
43
Print Extrapolated Offset Vector CAPCPP
45
Print Matrix AM*C Using Subroutine PRTL
Print Matrices LAM*C and PI*C Using Subroutine PRTL
A

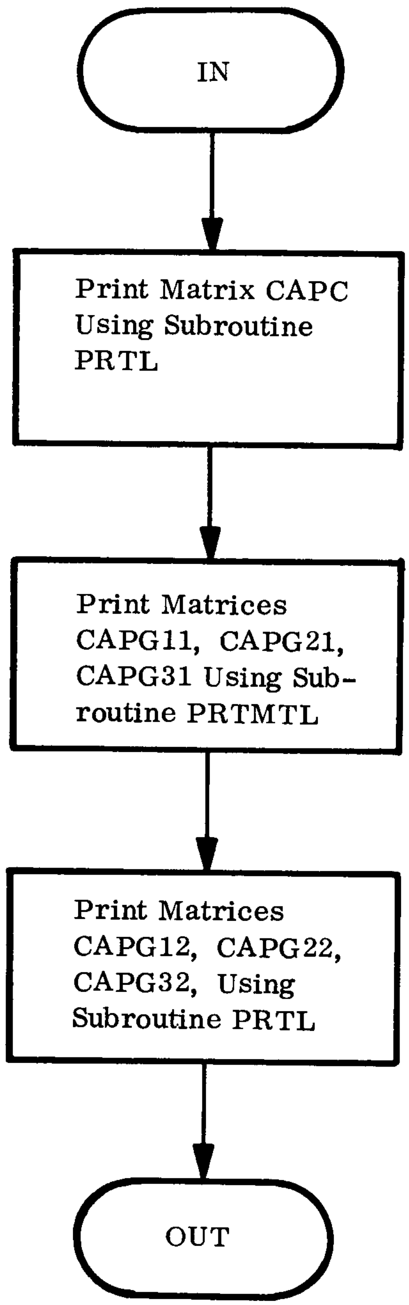
IGID


A
Rank (7)
< 3
OUT
≥3
Print Matrices PID*C, PHIC, C-1, BC, C-1, CC, C-1, GAMC, C-1 using subroutine PRTL
Print CGAM and CPHI
OUT

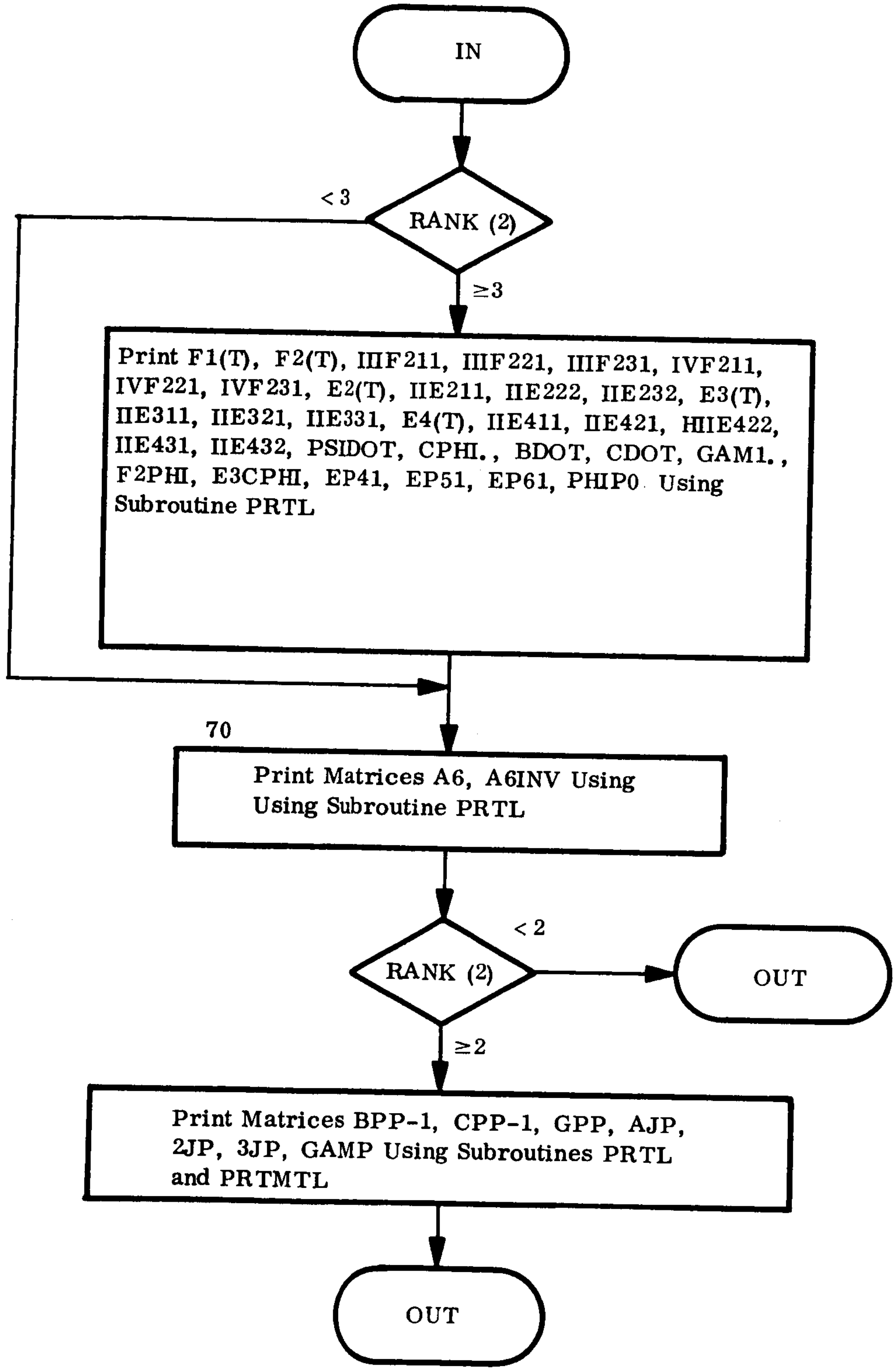
IGID ~ 2


IN
Print Control Weighting (WU) Matrix Using Subroutine PRTL
Print State Weighting (WX) Matrix Using Subrouting PRTL
OUT

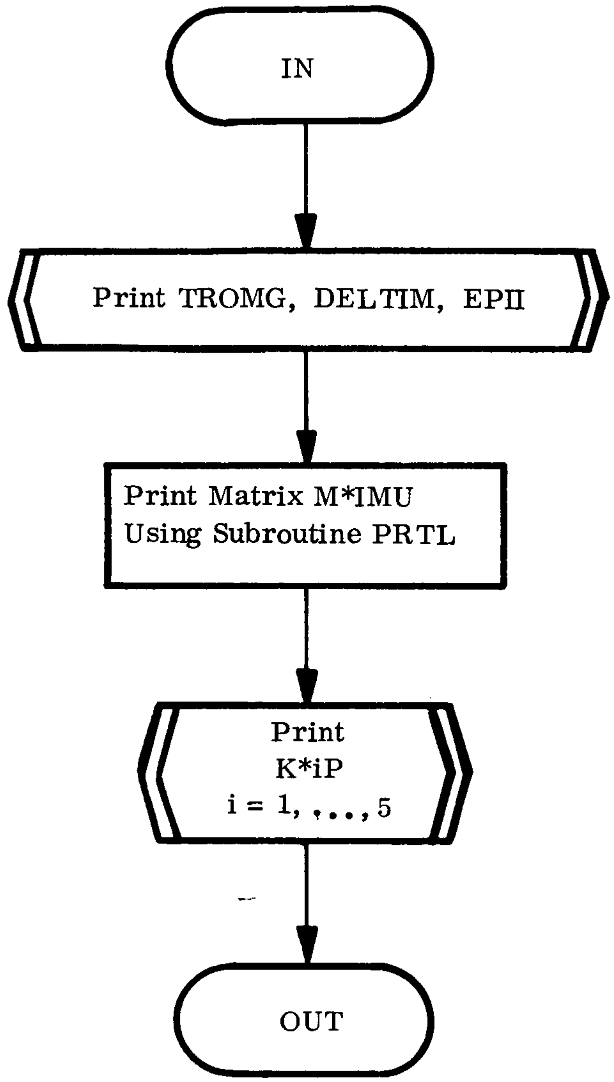
IGILA


IN
Print Matrix CAPC Using Subroutine PRTL
Print Matrices CAPG11, CAPG21, CAPG31 Using Sub-routine PRTMTL
Print Matrices CAPG12, CAPG22, CAPG32, Using Subroutine PRTL
OUT

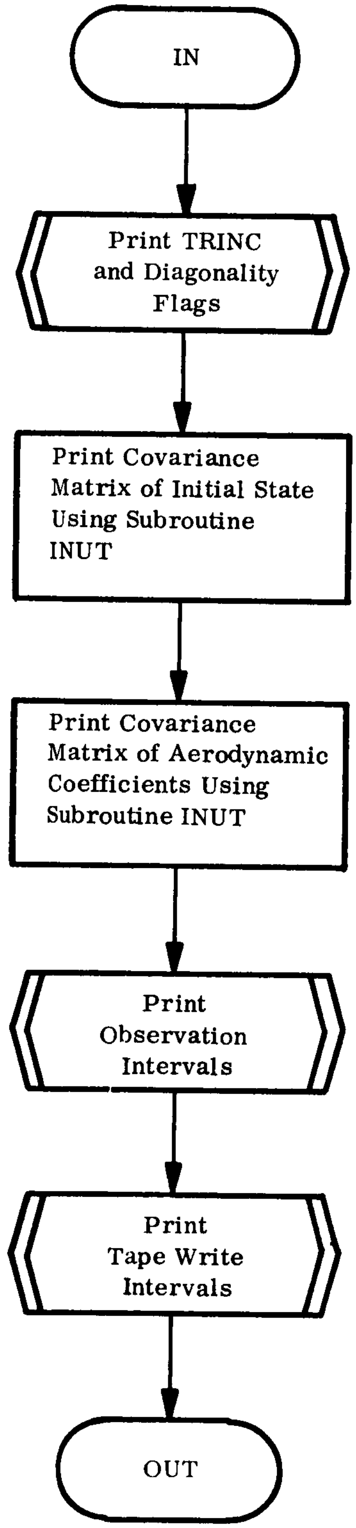
IIMU


IN
RANK (2)
< 3
≥3
Print F1(T), F2(T), IIIF211, IIIF221, IIIF231, IVF211, IVF221, IVF231, E2(T), IIE211, IIE222, IIE232, E3(T), IIE311, IIE321, IIE331, E4(T), IIE411, IIE421, HIIE422, IIE431, IIE432, PSIDOT, CPHI., BDOT, CDOT, GAM1., F2PHI, E3CPHI, EP41, EP51, EP61, PHIP0 Using Subroutine PRTL
70
Print Matrices A6, A6INV Using Using Subroutine PRTL
RANK (2)
< 2
OUT
≥2
Print Matrices BPP-1, CPP-1, GPP, AJP, 2JP, 3JP, GAMP Using Subroutines PRTL and PRTMTL
OUT

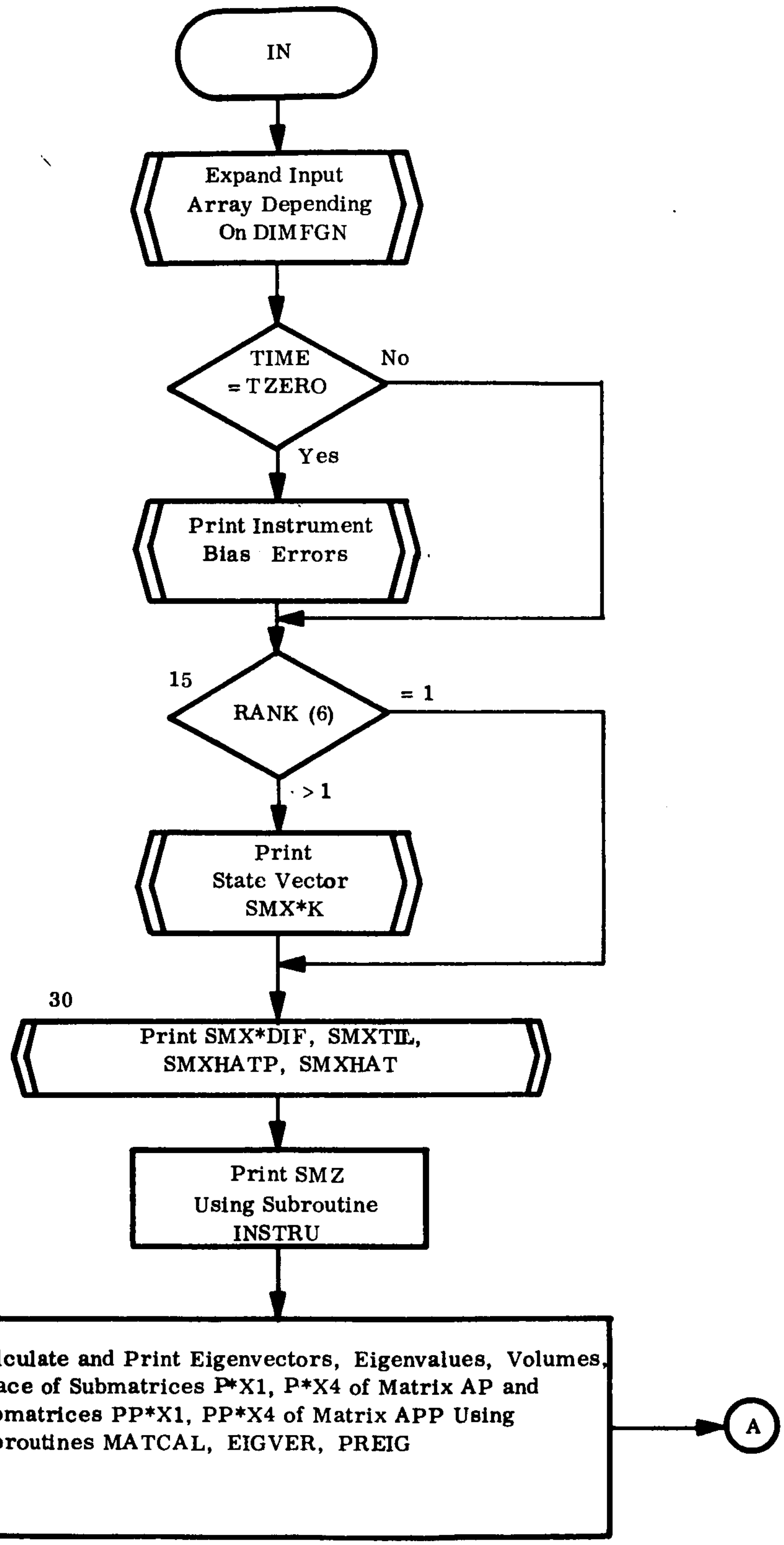
ILISYS


IN
Print TROMG, DELTIM, EPII
Print Matrix M*IMU
Using Subroutine PRTL
Print
K*iP
i = 1, . . . , 5
OUT

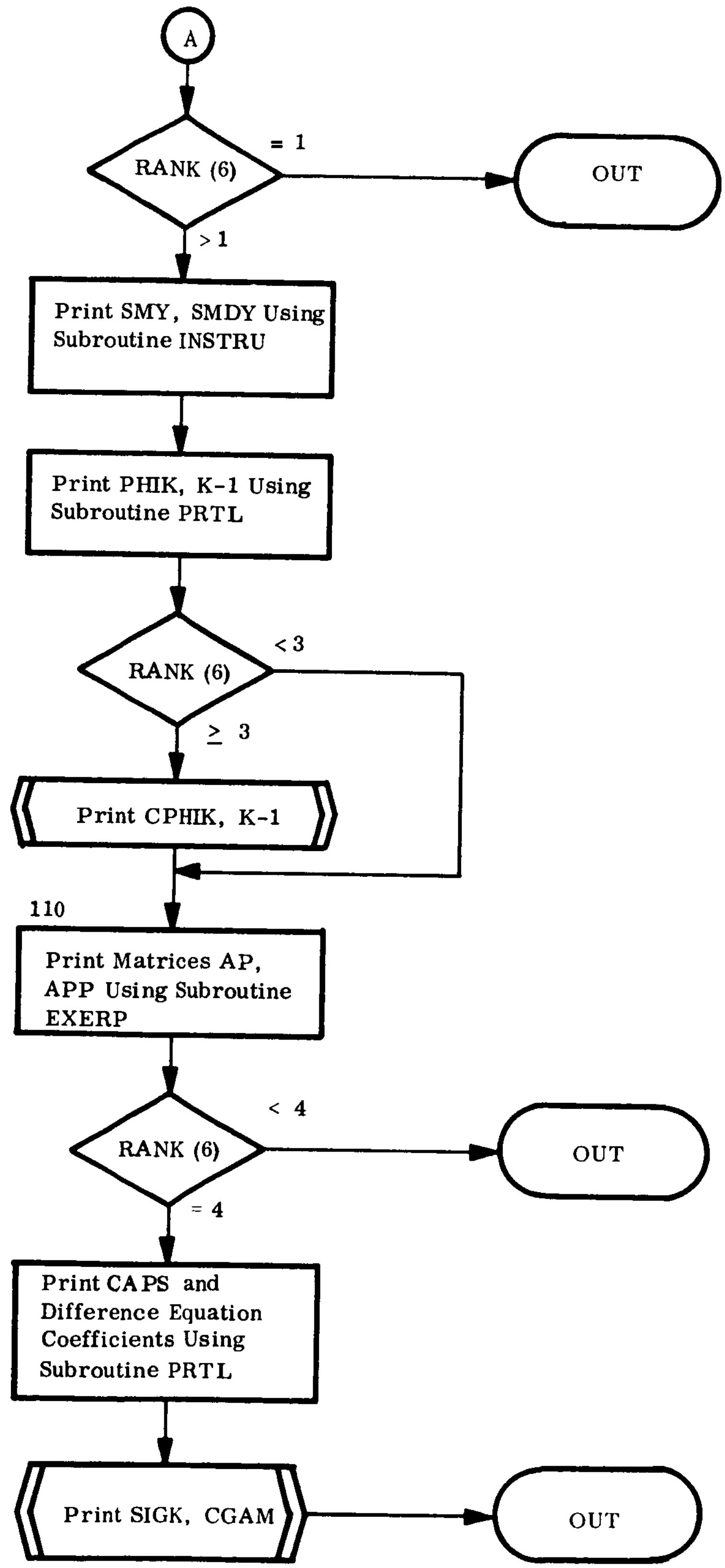
IMUIN


IN
Print TRINC and Diagonality Flags
Print Covariance Matrix of Initial State Using Subroutine INUT
Print Covariance Matrix of Aerodynamic Coefficients Using Subroutine INUT
Print Observation Intervals
Print Tape Write Intervals
OUT

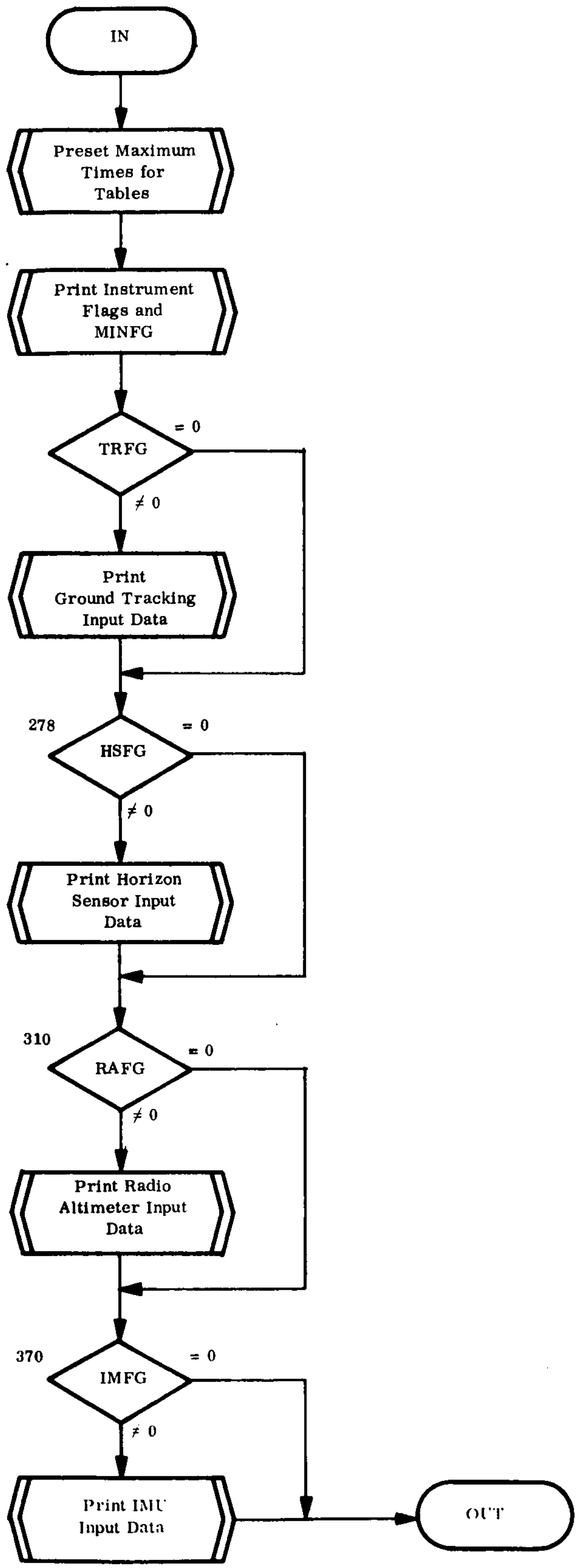
INACT


IN
Expand Input Array Depending On DIMFGN
TIME = TZERO
No
Yes
Print Instrument Bias Errors
15
RANK (6)
= 1
> 1
Print State Vector SMX*K
30
Print SMX*DIF, SMXTIL, SMXHATP, SMXHAT
Print SMZ Using Subroutine INSTRU
Calculate and Print Eigenvectors, Eigenvalues, Volumes, Trace of Submatrices P*X1, P*X4 of Matrix AP and Submatrices PP*X1, PP*X4 of Matrix APP Using Subroutines MATCAL, EIGVER, PREIG
A

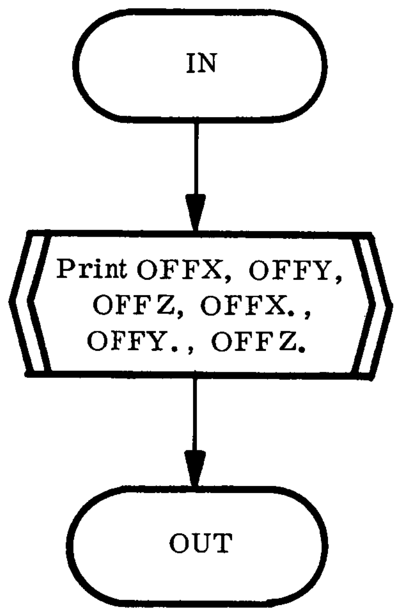
INAV


A
RANK (6)
= 1
OUT
> 1
Print SMY, SMDY Using Subroutine INSTRU
Print PHIK, K-1 Using Subroutine PRTL
RANK (6)
< 3
≥ 3
Print CPHIK, K-1
110
Print Matrices AP, APP Using Subroutine EXERP
RANK (6)
< 4
OUT
= 4
Print CAPS and Difference Equation Coefficients Using Subroutine PRTL
Print SIGK, CGAM
OUT

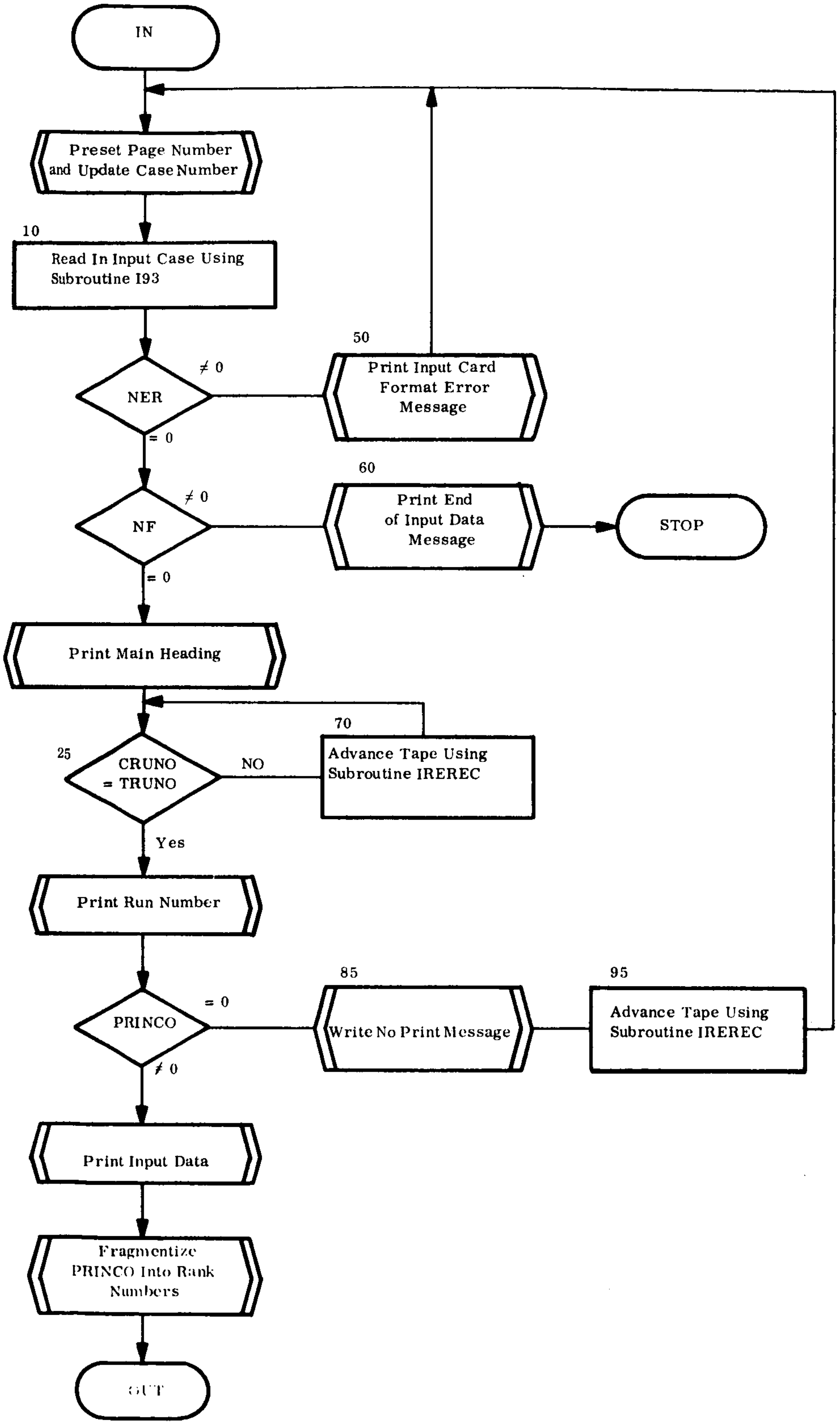
INAV~2


IN
Preset Maximum Times for Tables
Print Instrument Flags and MINFG
TRFG
= 0
≠ 0
Print Ground Tracking Input Data
278
HSFG
= 0
≠ 0
Print Horizon Sensor Input Data
310
RAFG
= 0
≠ 0
Print Radio Altimeter Input Data
370
IMFG
= 0
≠ 0
Print IMU Input Data
OUT

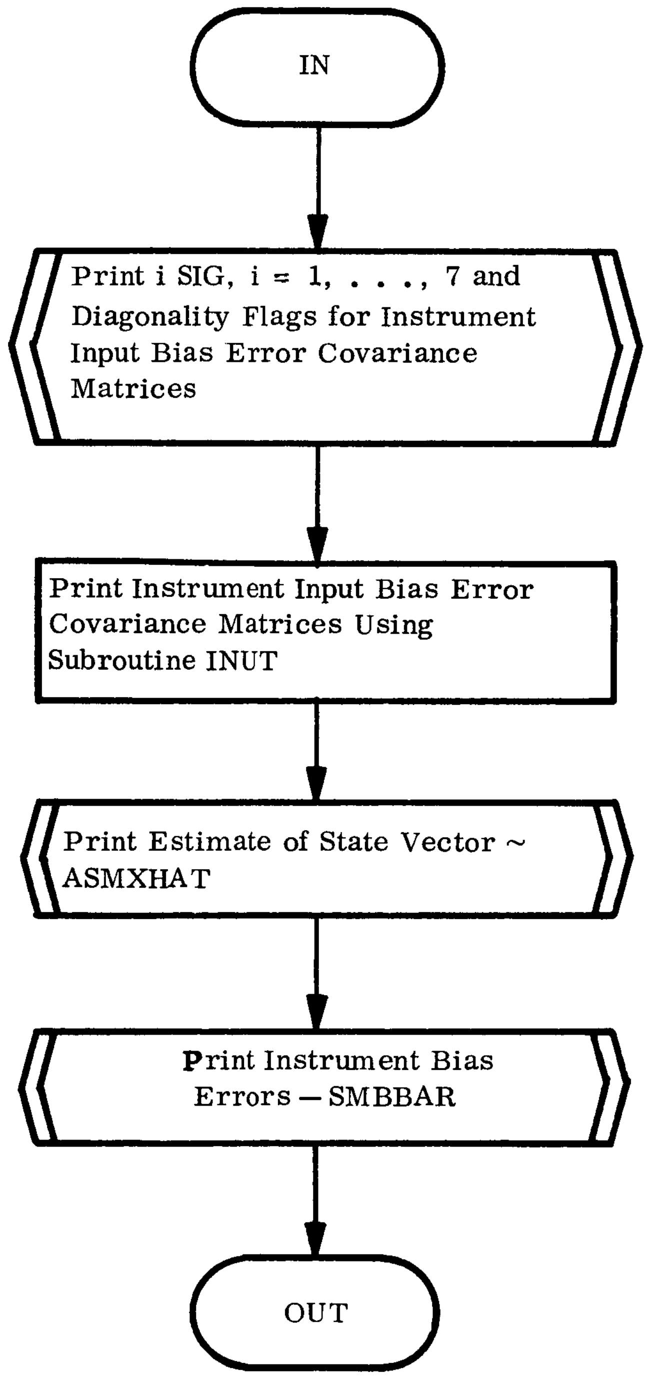
INEMS



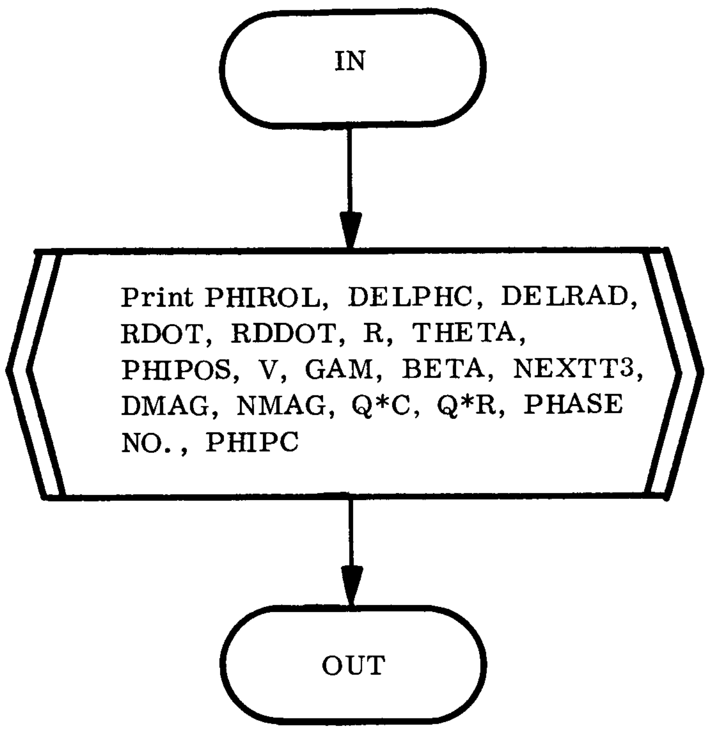
INGID


IN
Preset Page Number and Update Case Number
10
Read In Input Case Using Subroutine I93
50
Print Input Card Format Error Message
NER
≠ 0
= 0
60
Print End of Input Data Message
STOP
NF
≠ 0
= 0
Print Main Heading
70
Advance Tape Using Subroutine IREREC
25
CRUNO = TRUNO
NO
Yes
Print Run Number
85
Write No Print Message
95
Advance Tape Using Subroutine IREREC
PRINCO
= 0
≠ 0
Print Input Data
Fragmentize PRINCO Into Rank Numbers
OUT

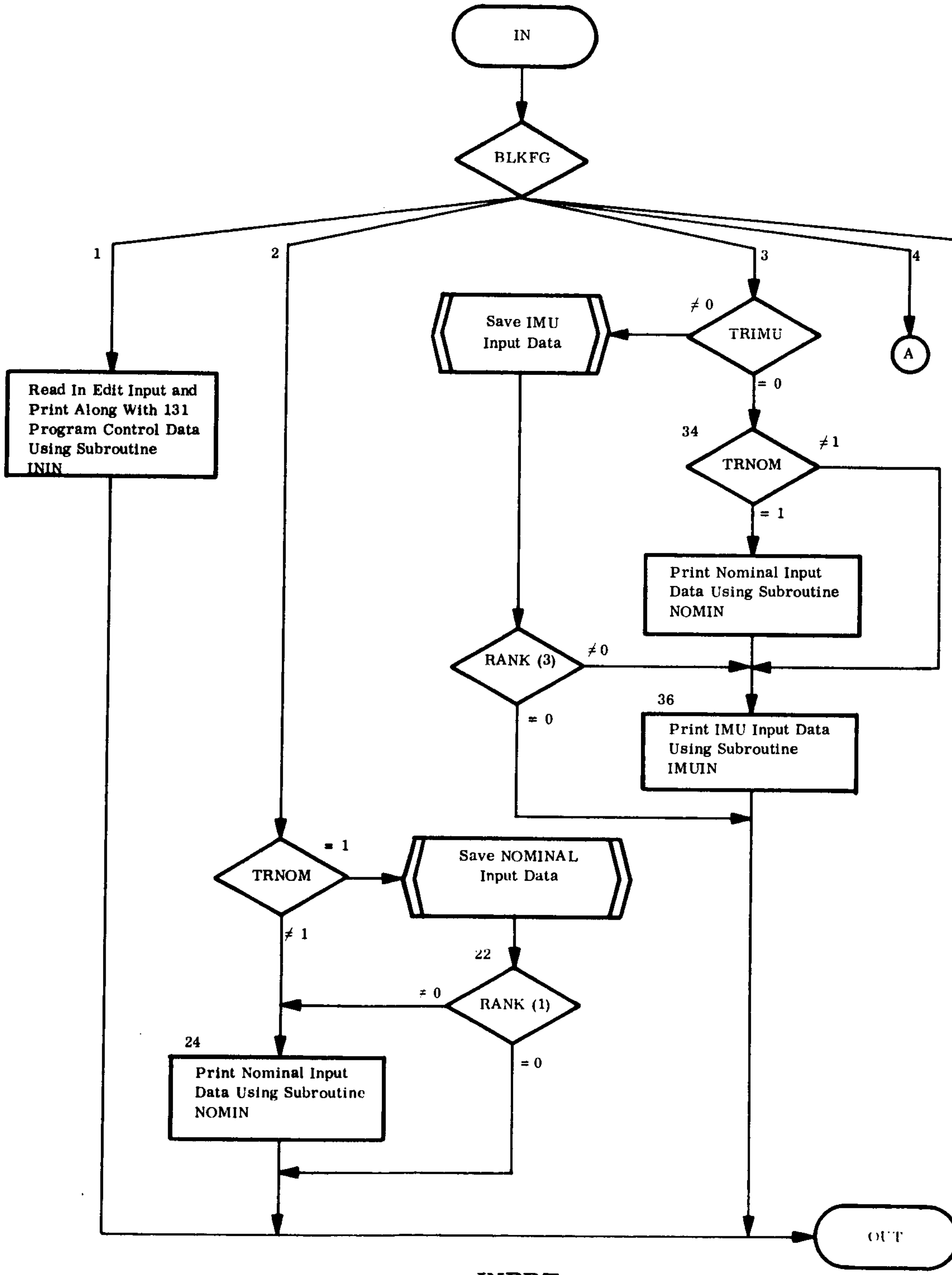
ININ


IN
Print i SIG, i = 1, ..., 7 and Diagonality Flags for Instrument Input Bias Error Covariance Matrices
Print Instrument Input Bias Error Covariance Matrices Using Subroutine INUT
Print Estimate of State Vector ~ ASMXHAT
Print Instrument Bias Errors — SMBBAR
OUT

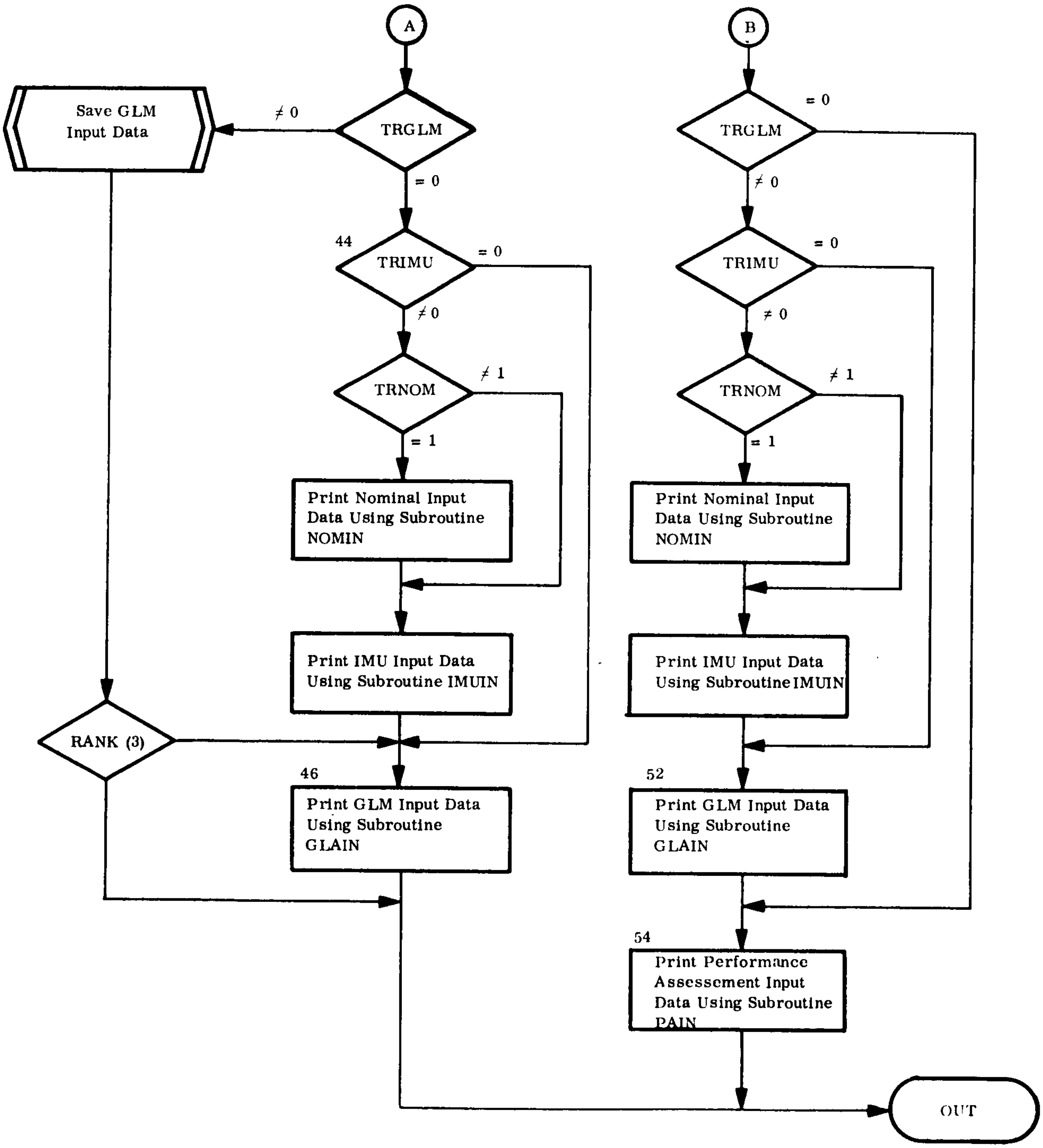
INNAV


IN
Print PHIROL, DELPHC, DELRAD, RDOT, RDDOT, R, THETA, PHIPOS, V, GAM, BETA, NEXTT3, DMAG, NMAG, Q*C, Q*R, PHASE NO., PHIPC
OUT

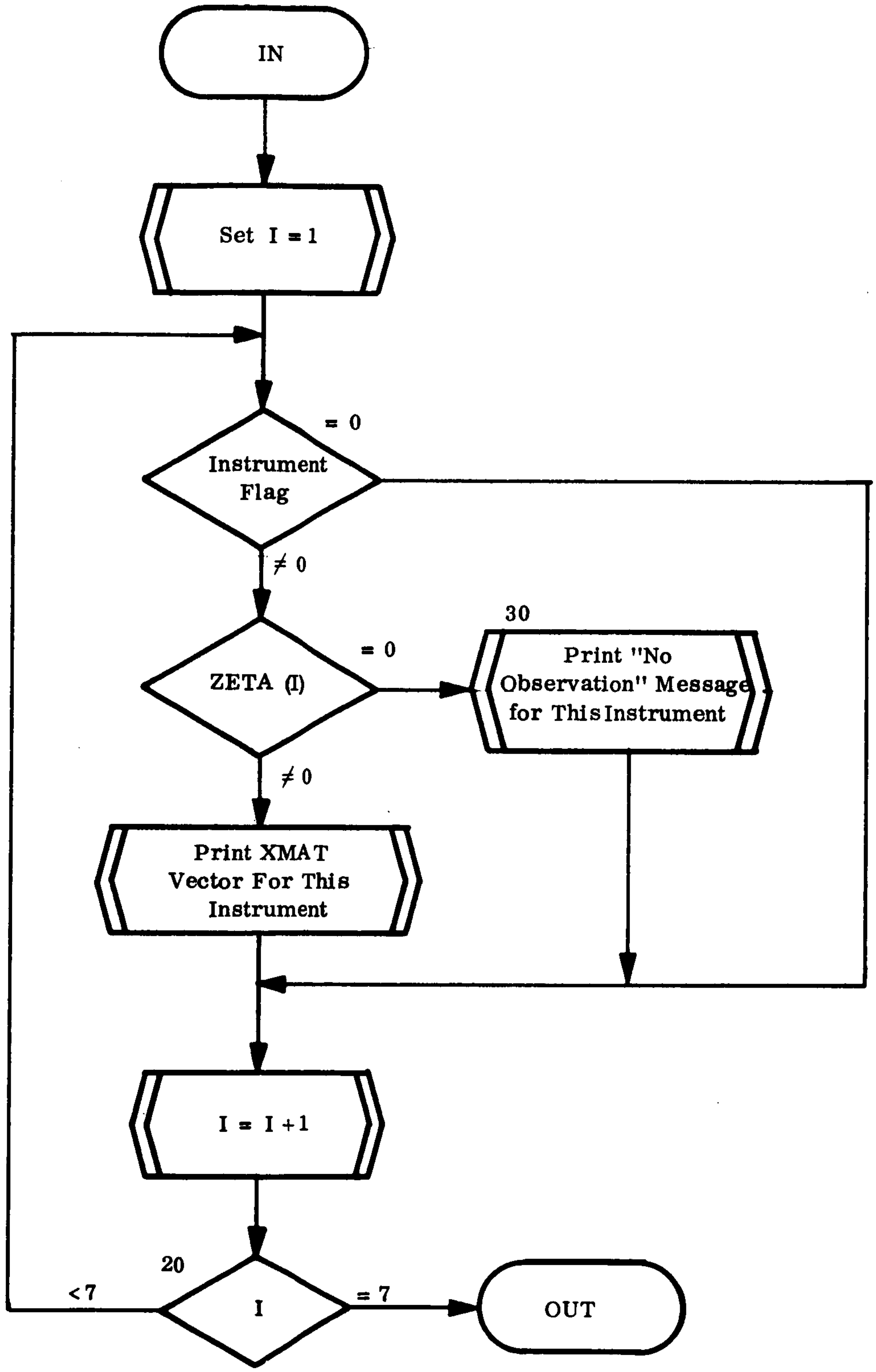
INOTRA


IN
BLKFG
1
2
3
4
5
≠ 0
TRIMU
Save IMU Input Data
A
B
Read In Edit Input and Print Along With 131 Program Control Data Using Subroutine ININ
= 0
34
TRNOM
≠ 1
= 1
Print Nominal Input Data Using Subroutine NOMIN
RANK (3)
≠ 0
36
= 0
Print IMU Input Data Using Subroutine IMUIN
= 1
TRNOM
Save NOMINAL Input Data
≠ 1
22
≠ 0
RANK (1)
24
= 0
Print Nominal Input Data Using Subroutine NOMIN
OUT

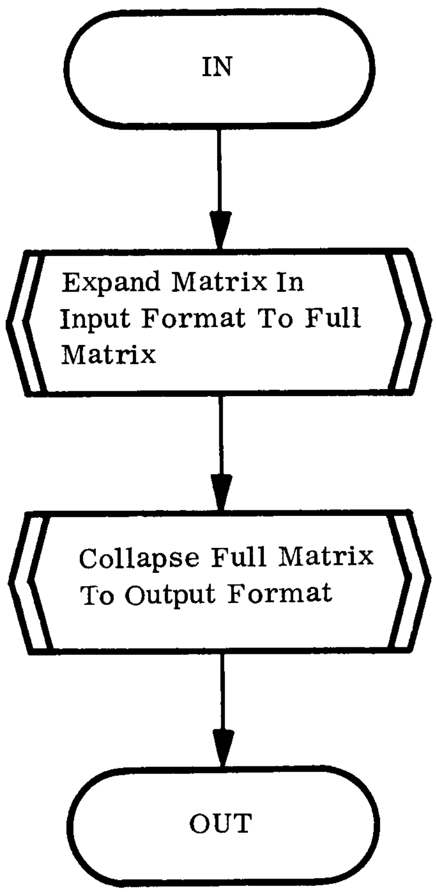
INPRT


A
B
Save GLM Input Data
≠ 0
TRGLM
= 0
44
TRIMU
= 0
≠ 0
TRNOM
≠ 1
= 1
Print Nominal Input Data Using Subroutine NOMIN
Print IMU Input Data Using Subroutine IMUIN
RANK (3)
46
Print GLM Input Data Using Subroutine GLAIN
52
54
Print Performance Assessement Input Data Using Subroutine PAIN
OUT

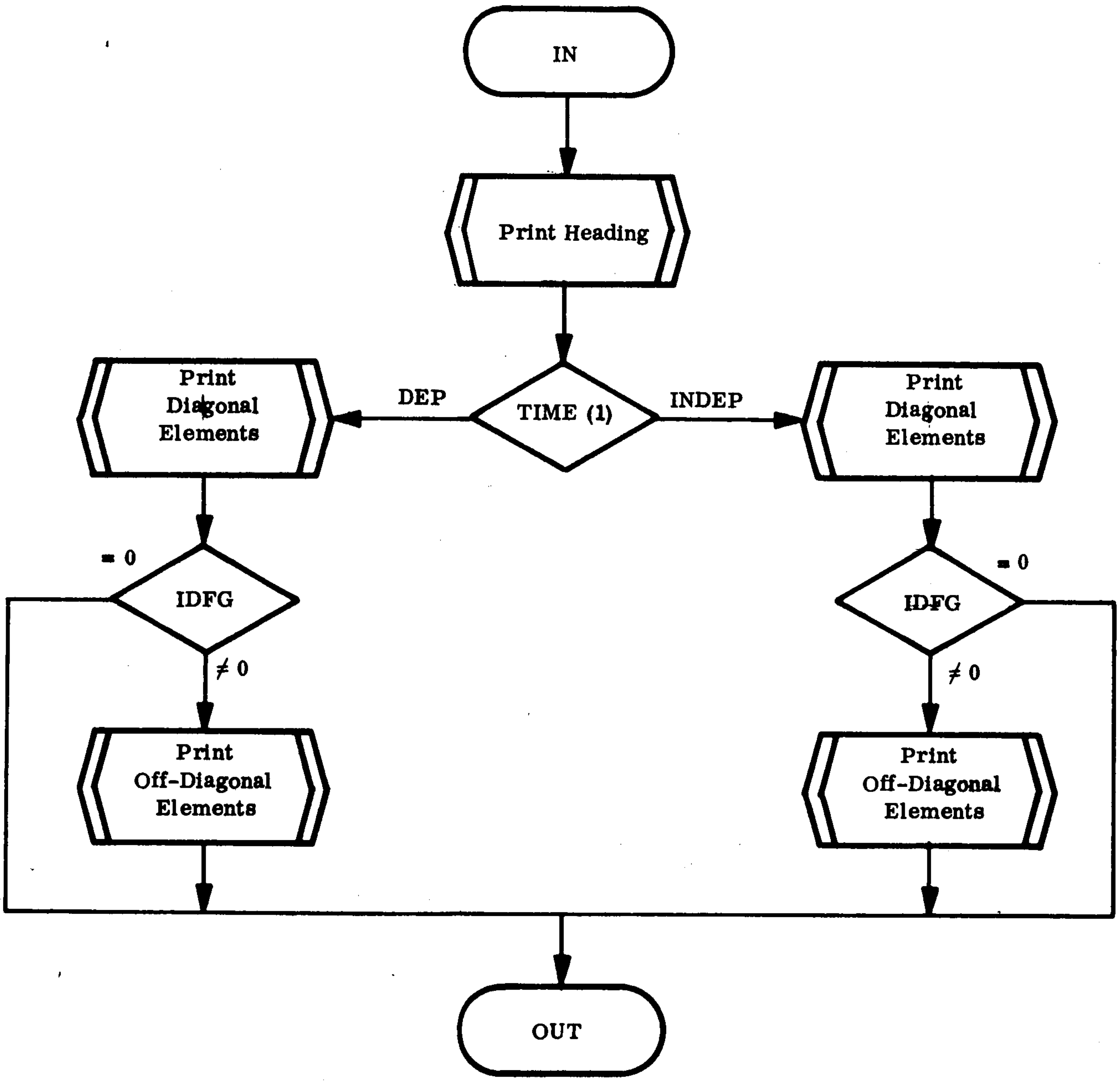
INPRT ~2


IN
Set I = 1
= 0
Instrument Flag
≠ 0
30
= 0
ZETA (I)
Print "No Observation" Message for This Instrument
≠ 0
Print XMAT Vector For This Instrument
I = I + 1
20
< 7
I
= 7
OUT

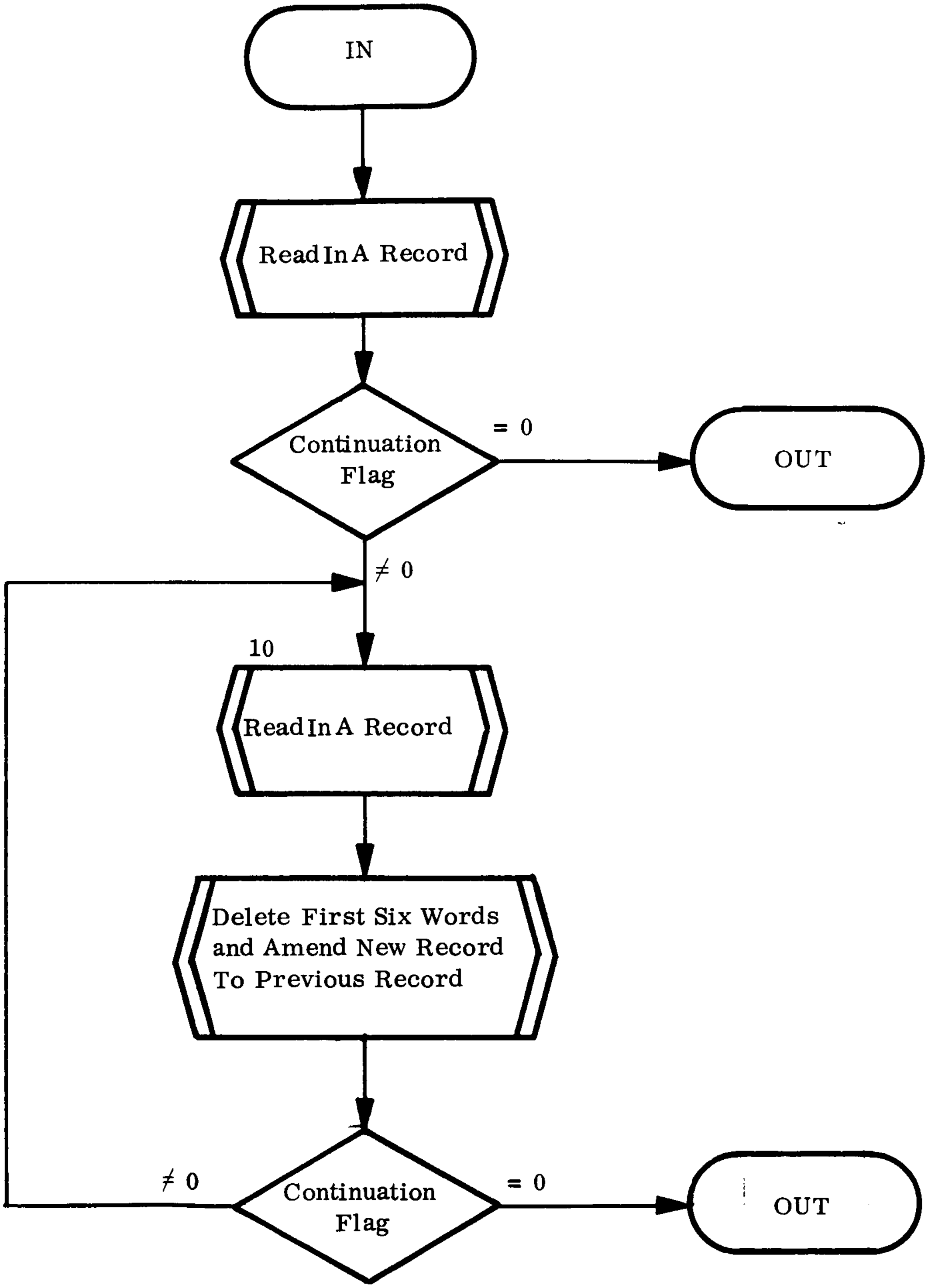
INSTRU


IN
Expand Matrix In Input Format To Full Matrix
Collapse Full Matrix To Output Format
OUT

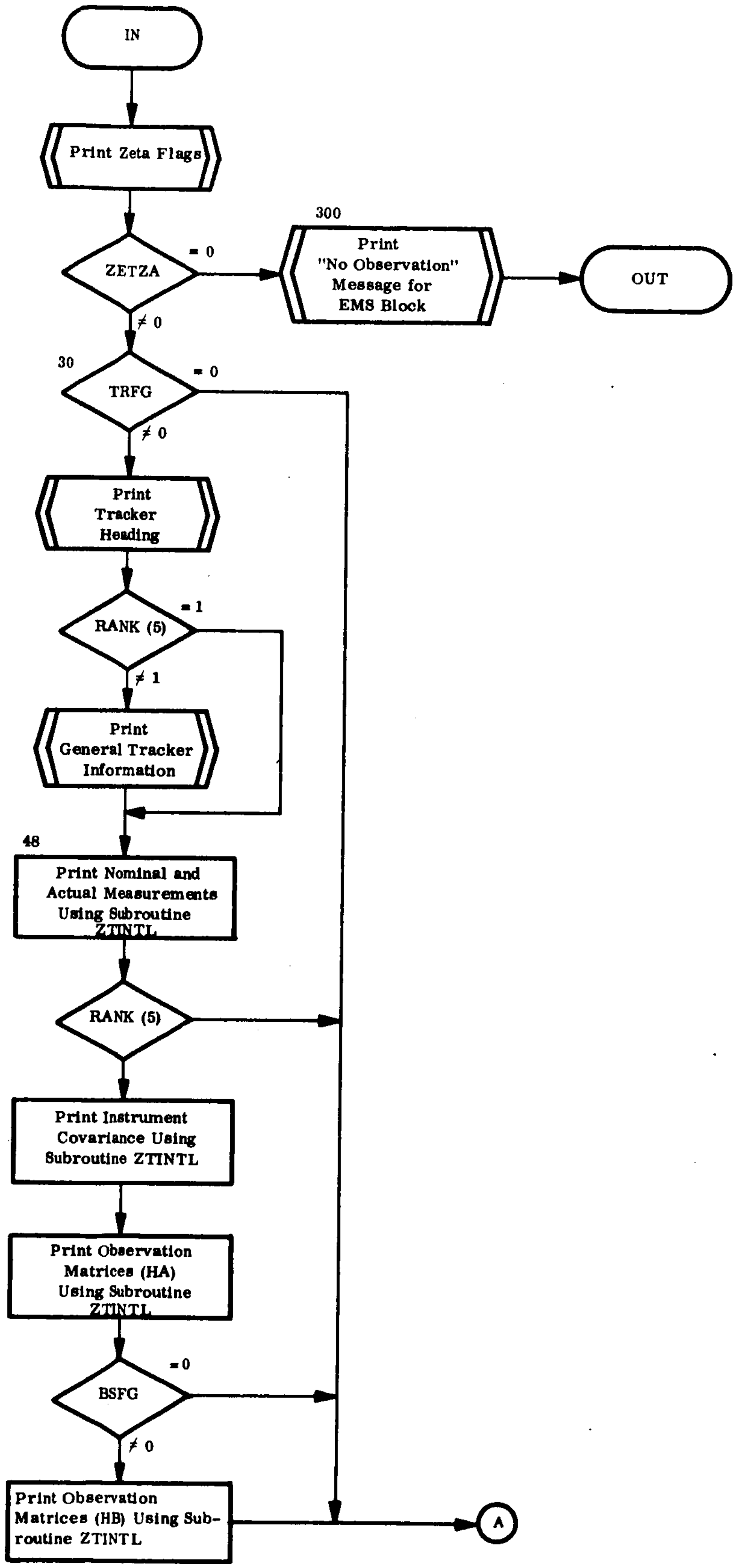
INUADJ


IN
Print Heading
TIME (1)
DEP
INDEP
Print Diagonal Elements
Print Diagonal Elements
IDFG
IDFG
= 0
≠ 0
= 0
≠ 0
Print Off-Diagonal Elements
Print Off-Diagonal Elements
OUT

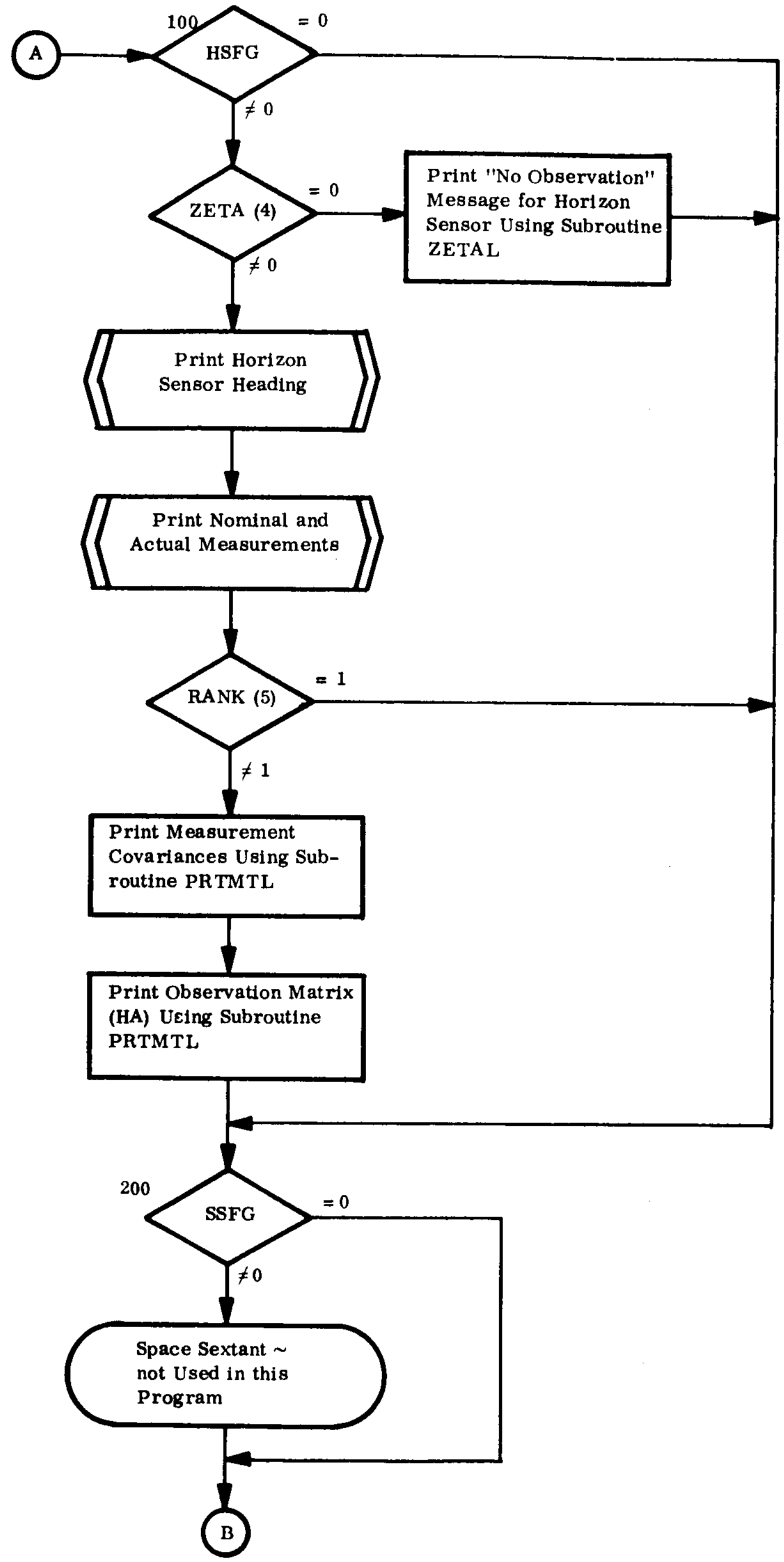
INUT


IN
Read In A Record
Continuation Flag
= 0
OUT
≠ 0
10
Read In A Record
Delete First Six Words and Amend New Record To Previous Record
≠ 0
Continuation Flag
= 0
OUT

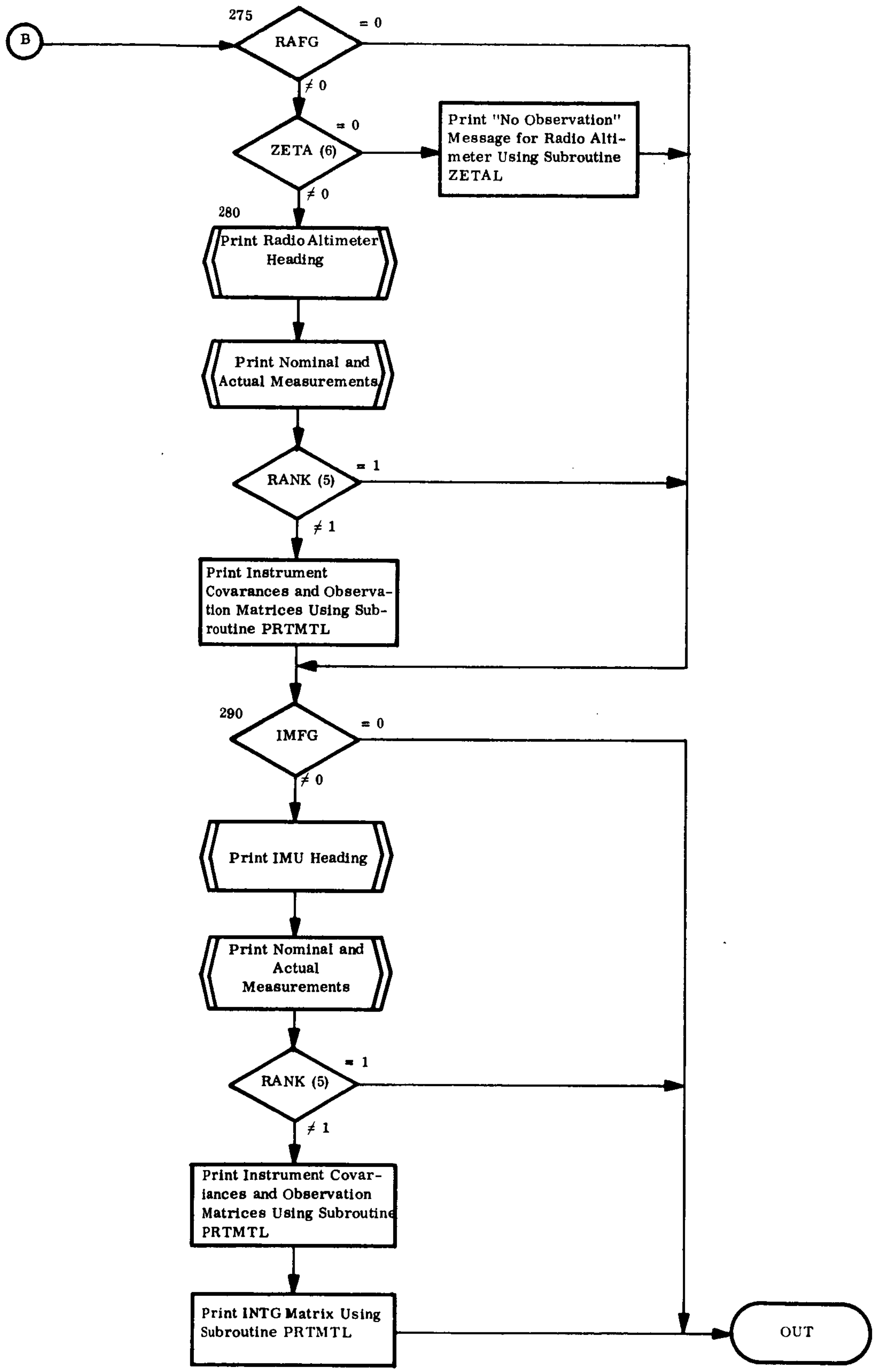
IREREC


IN
Print Zeta Flags
ZETZA
= 0
≠ 0
300
Print "No Observation" Message for EMS Block
OUT
30
TRFG
= 0
≠ 0
Print Tracker Heading
RANK (5)
= 1
≠ 1
Print General Tracker Information
48
Print Nominal and Actual Measurements Using Subroutine ZTINTL
RANK (5)
Print Instrument Covariance Using Subroutine ZTINTL
Print Observation Matrices (HA) Using Subroutine ZTINTL
BSFG
= 0
≠ 0
Print Observation Matrices (HB) Using Subroutine ZTINTL
A

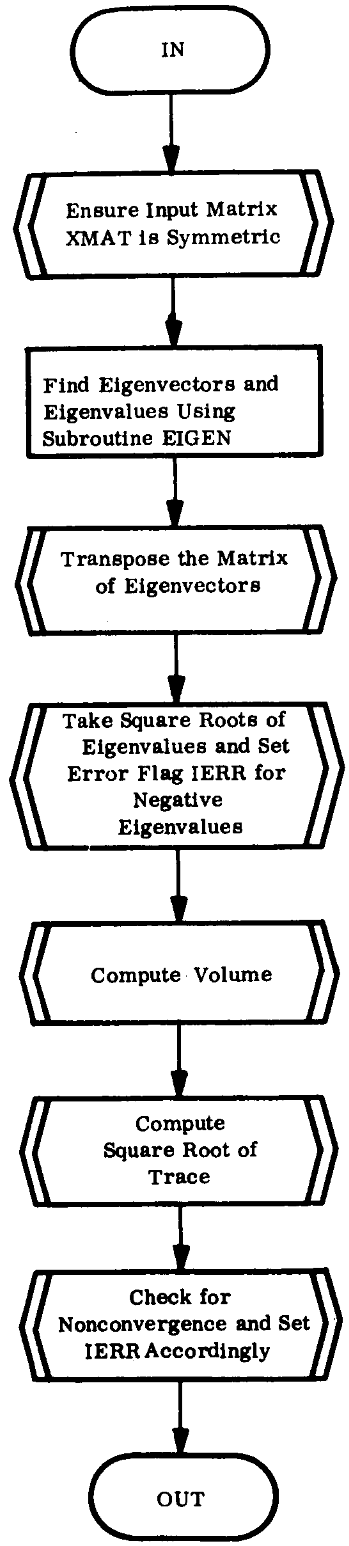
ITEMS


A
100
HSFG
= 0
≠ 0
ZETA (4)
= 0
Print "No Observation" Message for Horizon Sensor Using Subroutine ZETAL
≠ 0
Print Horizon Sensor Heading
Print Nominal and Actual Measurements
RANK (5)
= 1
≠ 1
Print Measurement Covariances Using Sub-routine PRTMTL
Print Observation Matrix (HA) Using Subroutine PRTMTL
200
SSFG
= 0
≠ 0
Space Sextant ~ not Used in this Program
B

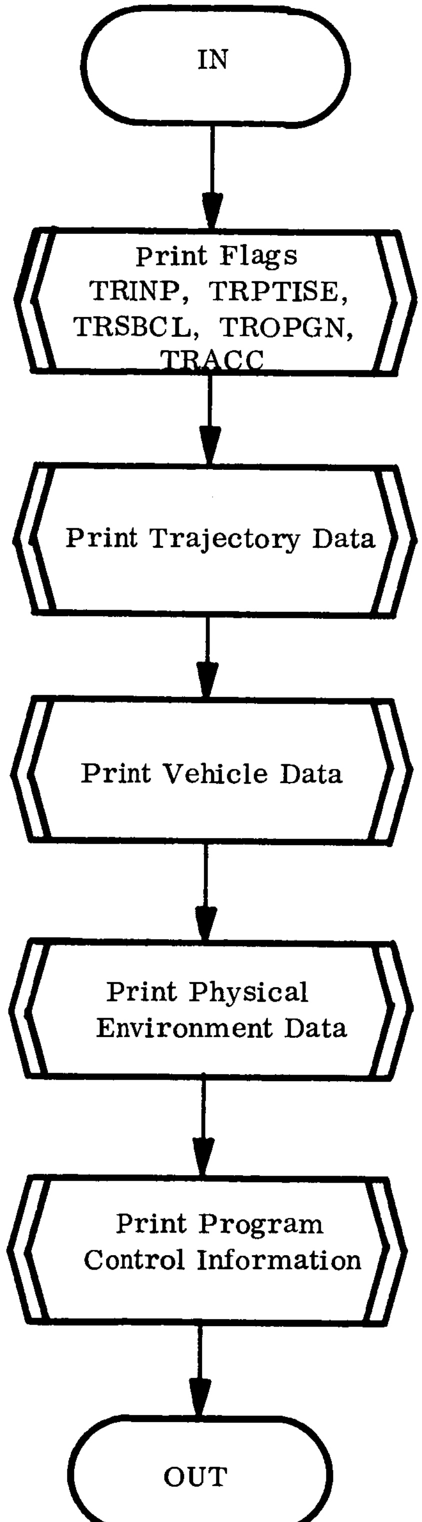
ITEMS ~ 2


B
275
RAFG
= 0
≠ 0
ZETA (6)
= 0
Print "No Observation" Message for Radio Altimeter Using Subroutine ZETAL
≠ 0
280
Print Radio Altimeter Heading
Print Nominal and Actual Measurements
RANK (5)
= 1
≠ 1
Print Instrument Covarances and Observation Matrices Using Subroutine PRTMTL
290
IMFG
= 0
≠ 0
Print IMU Heading
Print Nominal and Actual Measurements
RANK (5)
= 1
≠ 1
Print Instrument Covariances and Observation Matrices Using Subroutine PRTMTL
Print INTG Matrix Using Subroutine PRTMTL
OUT

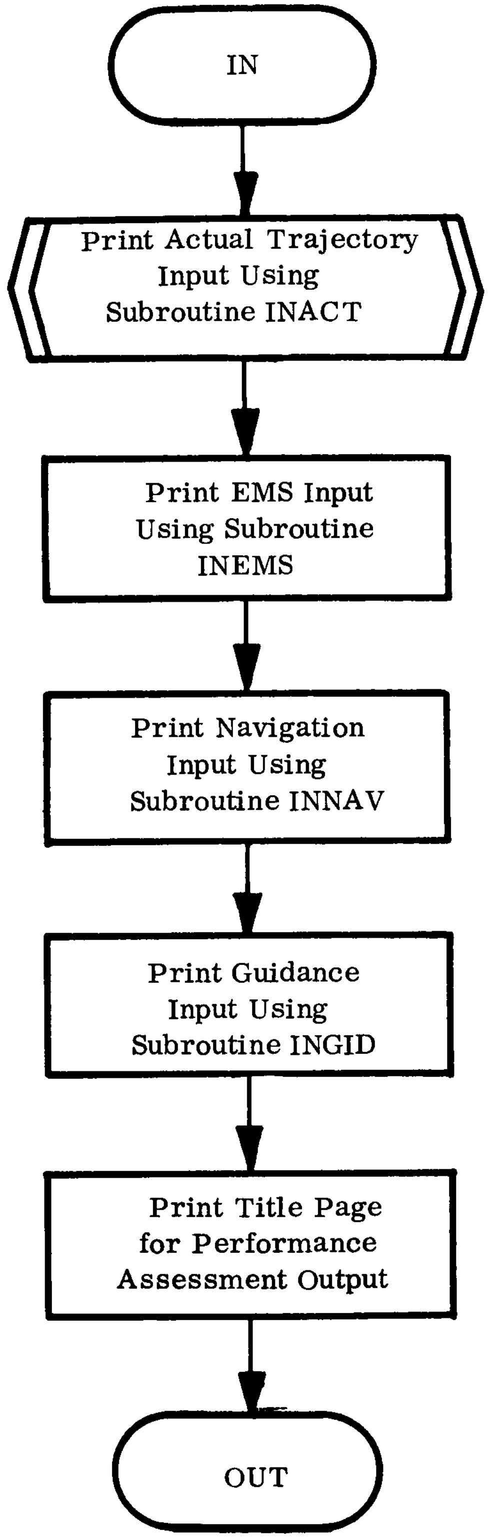
ITEMS ~ 3


IN
Ensure Input Matrix XMAT is Symmetric
Find Eigenvectors and Eigenvalues Using Subroutine EIGEN
Transpose the Matrix of Eigenvectors
Take Square Roots of Eigenvalues and Set Error Flag IERR for Negative Eigenvalues
Compute Volume
Compute Square Root of Trace
Check for Nonconvergence and Set IERR Accordingly
OUT

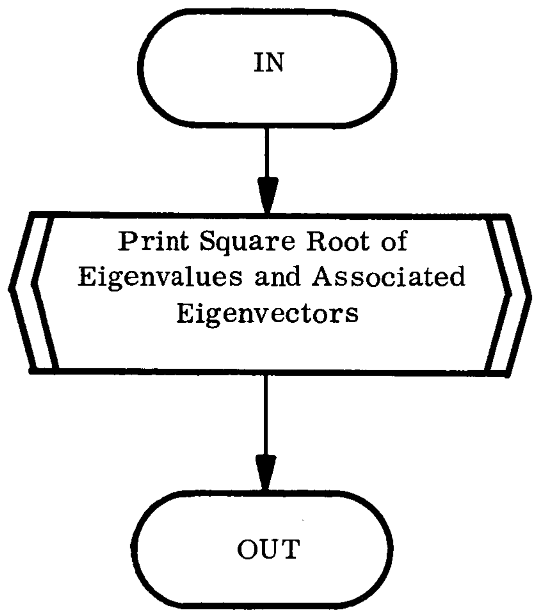
MATCAL


IN
Print Flags
TRINP, TRPTISE,
TRSBCL, TROPGN,
TRACC
Print Trajectory Data
Print Vehicle Data
Print Physical
Environment Data
Print Program
Control Information
OUT

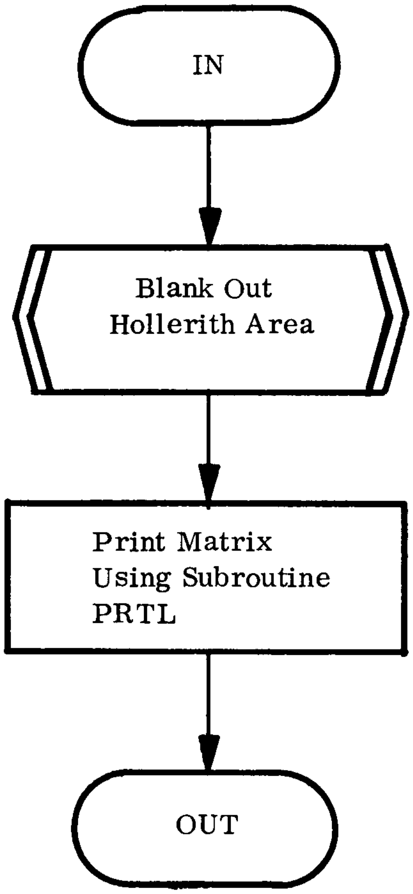
NOMIN


IN
Print Actual Trajectory Input Using Subroutine INACT
Print EMS Input Using Subroutine INEMS
Print Navigation Input Using Subroutine INNAV
Print Guidance Input Using Subroutine INGID
Print Title Page for Performance Assessment Output
OUT

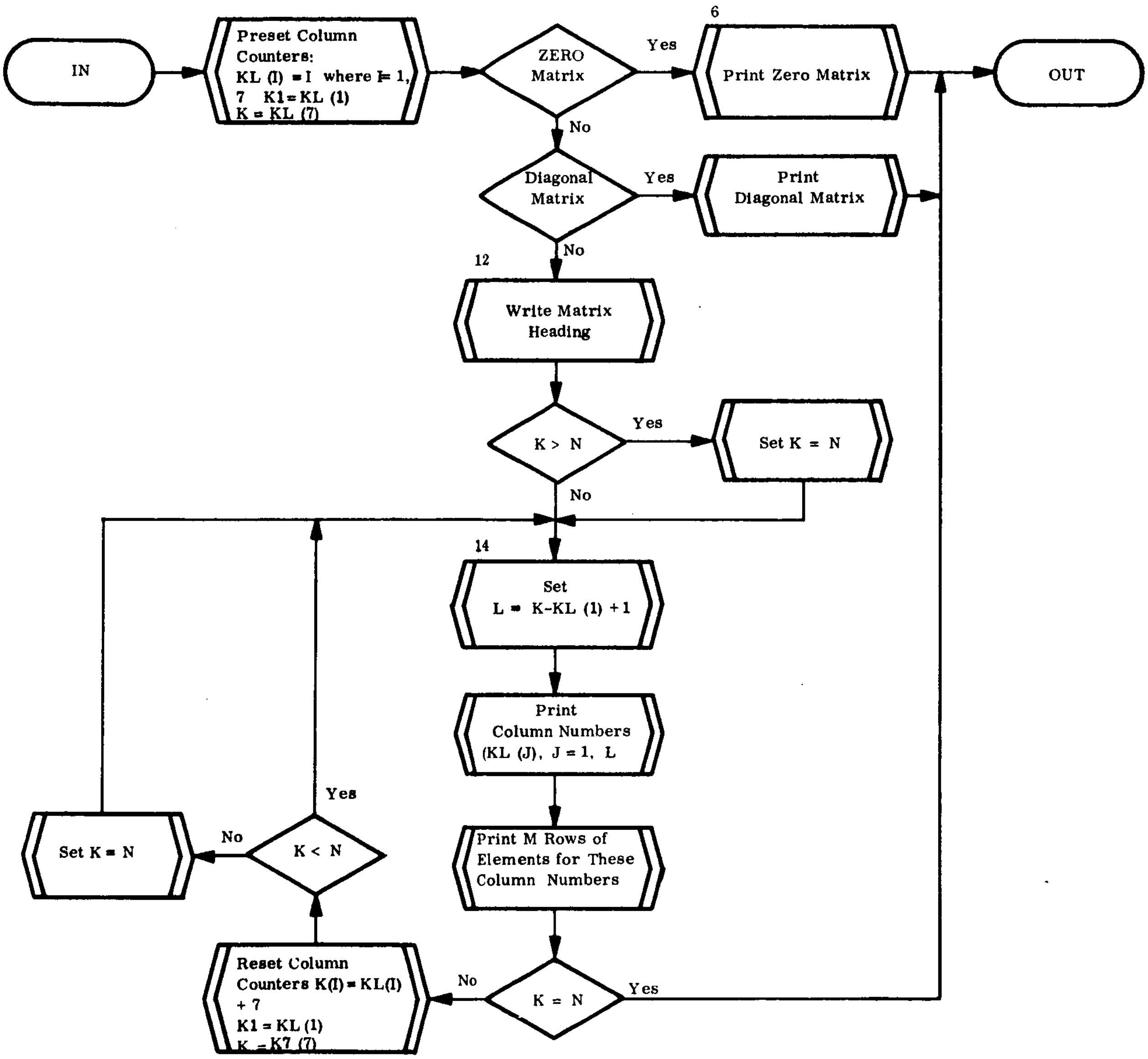
PAIN


IN
Print Square Root of Eigenvalues and Associated Eigenvectors
OUT

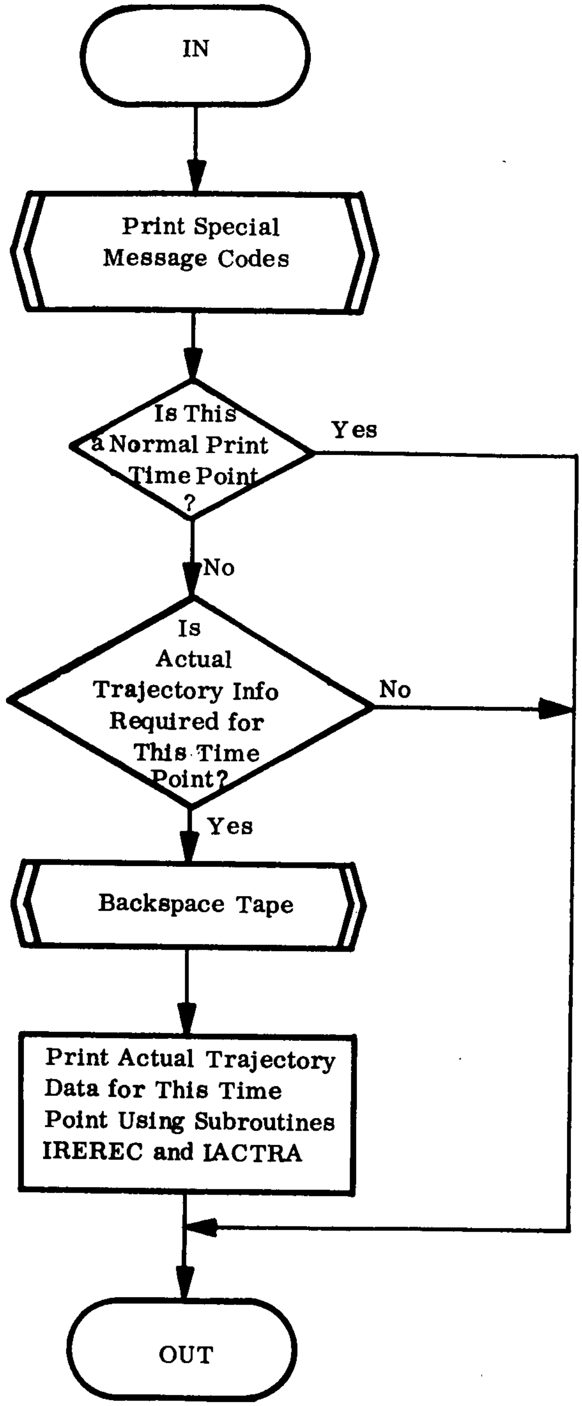
PREIG


IN
Blank Out
Hollerith Area
Print Matrix
Using Subroutine
PRTL
OUT

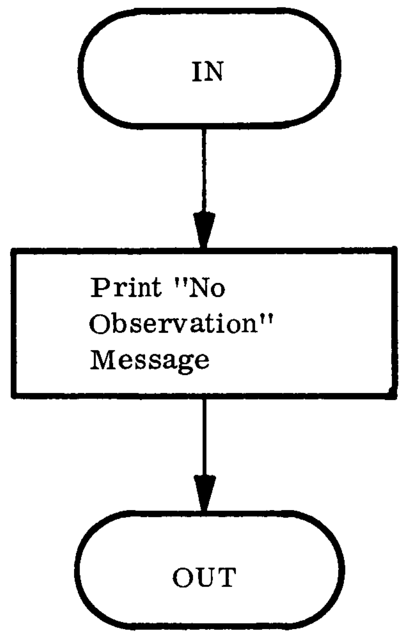
PRTMTL


IN
Preset Column Counters: KL (I) = I where I = 1, 7 K1 = KL (1) K = KL (7)
ZERO Matrix
Yes
No
6
Print Zero Matrix
OUT
Diagonal Matrix
Yes
No
Print Diagonal Matrix
12
Write Matrix Heading
K > N
Yes
No
Set K = N
14
Set L = K-KL (1) + 1
Print Column Numbers (KL (J), J = 1, L
Print M Rows of Elements for These Column Numbers
K = N
Yes
No
Reset Column Counters K(I) = KL(I) + 7 K1 = KL (1) K = K7 (7)
K < N
Yes
No
Set K = N

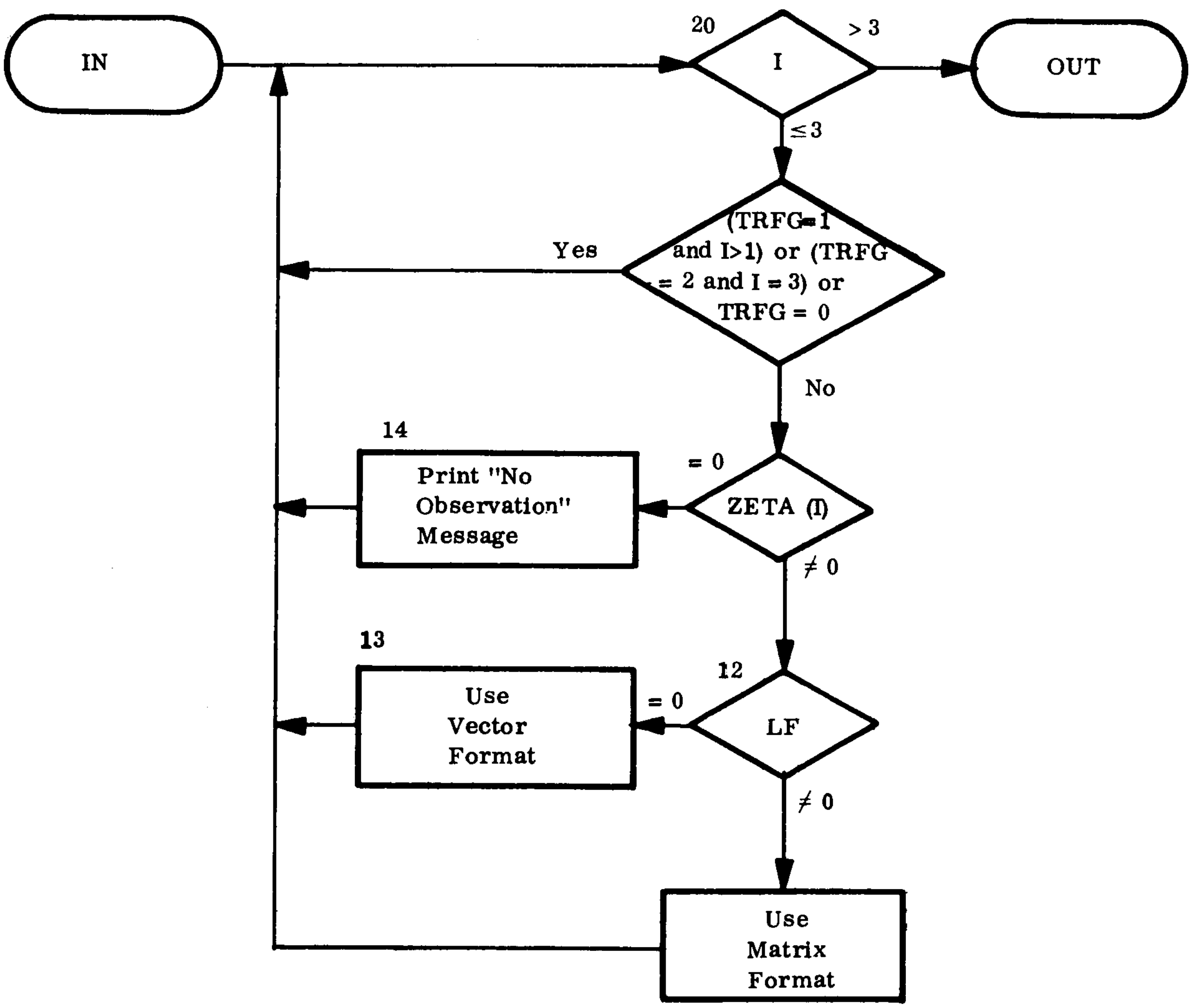
PRTL


IN
Print Special Message Codes
Is This a Normal Print Time Point ?
Yes
No
Is Actual Trajectory Info Required for This Time Point?
No
Yes
Backspace Tape
Print Actual Trajectory Data for This Time Point Using Subroutines IREREC and IACTRA
OUT

SPEMES



ZETAL

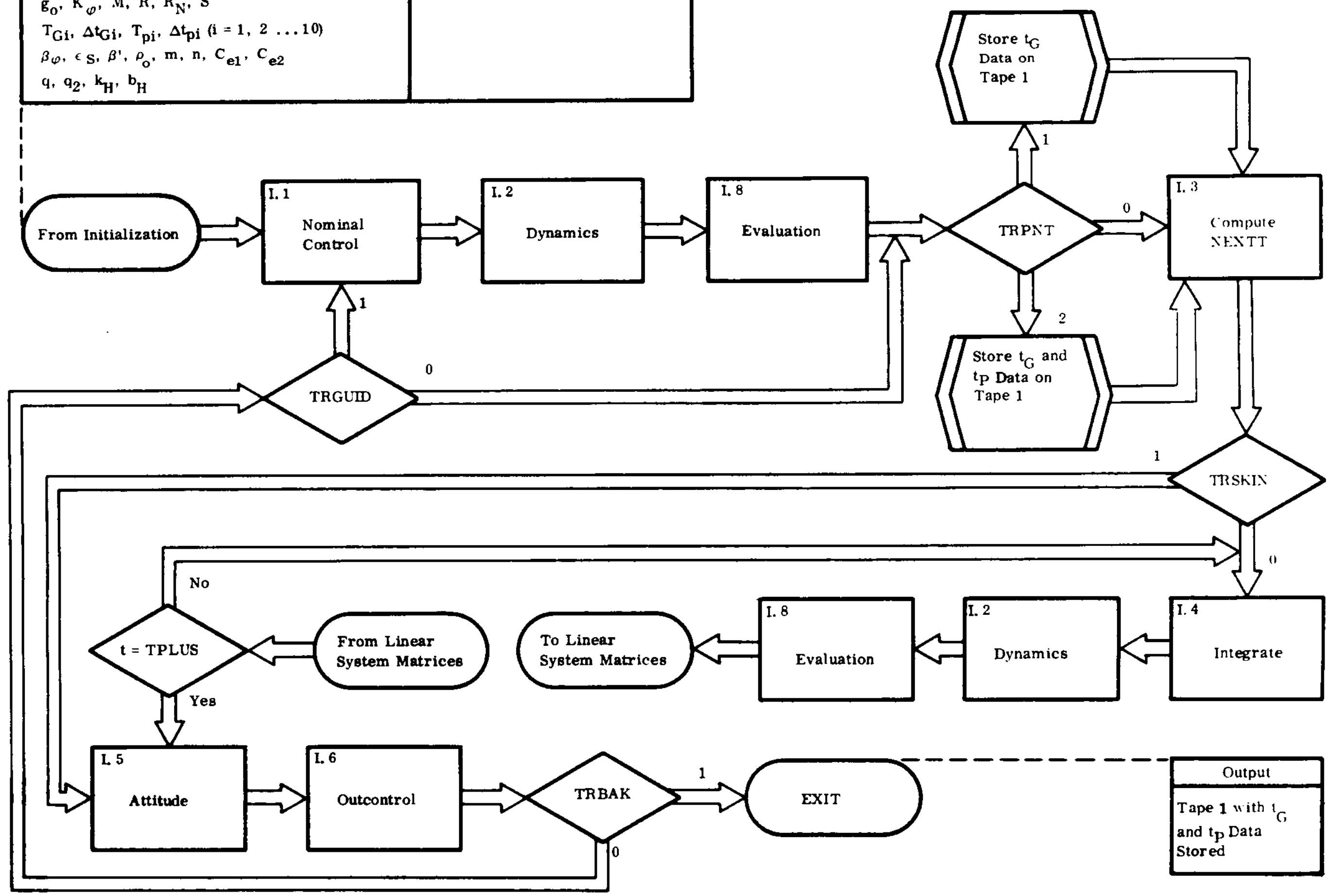

IN
20
I
> 3
OUT
≤ 3
(TRFG = 1 and I>1) or (TRFG = 2 and I = 3) or TRFG = 0
Yes
No
14
Print "No Observation" Message
= 0
ZETA (I)
≠ 0
13
Use Vector Format
12
= 0
LF
≠ 0
Use Matrix Format

ZTINTL

3.4 BASIC COMPUTATIONAL BLOCKS

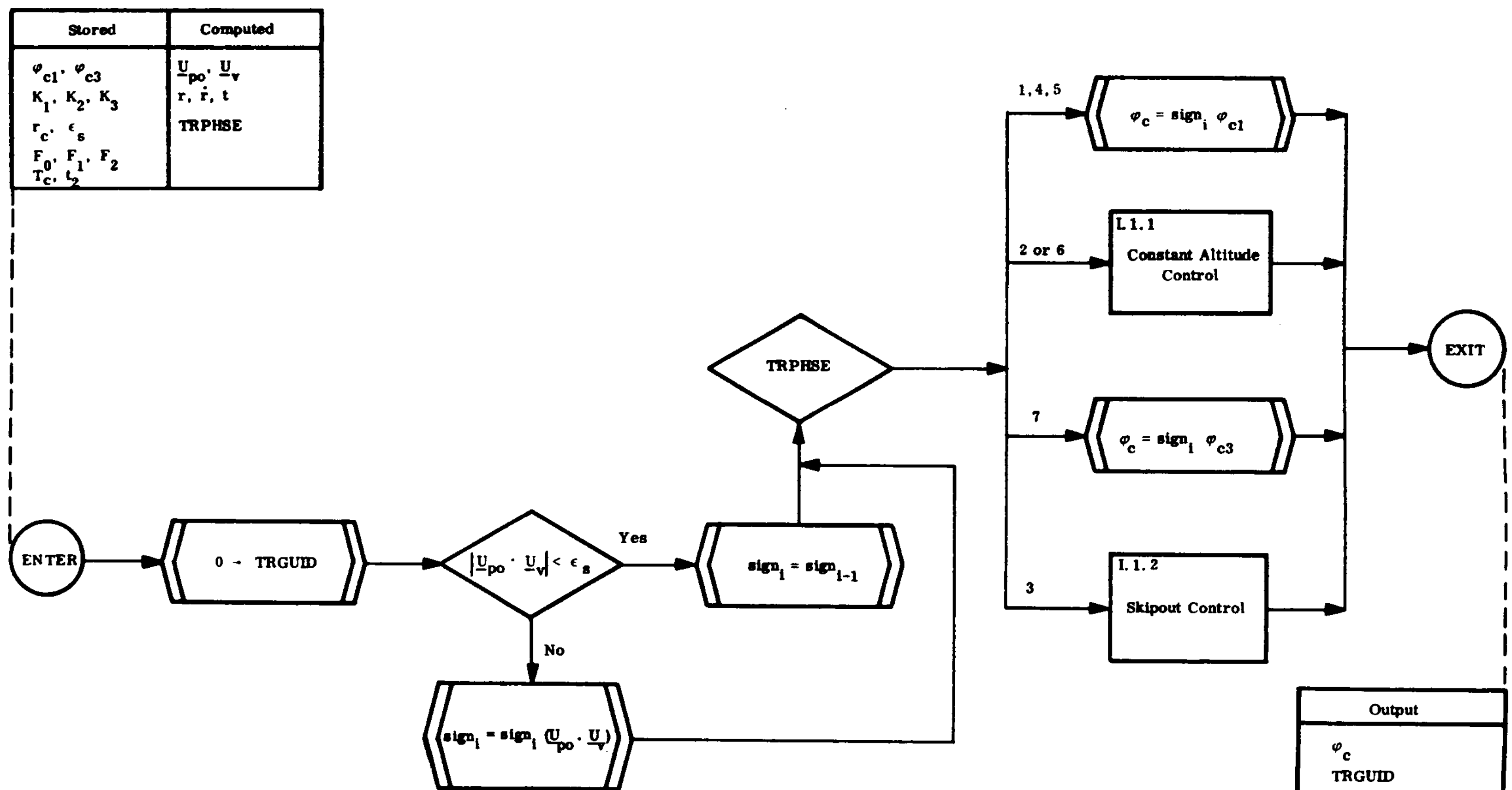
3.4.1 Nominal Trajectory

Stored	Computed
TRPHSE, TROPGN t_o, t_3, t_3', δt_1, ϵ_1, δt_2, ϵ_2, φ_{11}, φ_{21} F_{1i}, F_{2i} $(i = 0, 1, 2)$ K_i $(i = 1, 2, 3)$ or ζ, τ C_D, $C_{N\alpha}$, C_2, C_3, C_4, C_5, C_H g_o, K_φ, M, R, R_N, S T_{Gi}, Δt_{Gi}, T_{pi}, Δt_{pi} $(i = 1, 2 \ldots 10)$ β_φ, ϵ_S, β', ρ_o, m, n, C_{e1}, C_{e2} q, q_2, k_H, b_H	A_4 (3×3), A_5 (3×3) $\underline{r}_o$, $\underline{V}_o$ $\underline{P}_{Io}$, $\underline{Y}_{Ao}$, $\underline{R}_{oo}$ $\underline{U}_{po}$, r_c

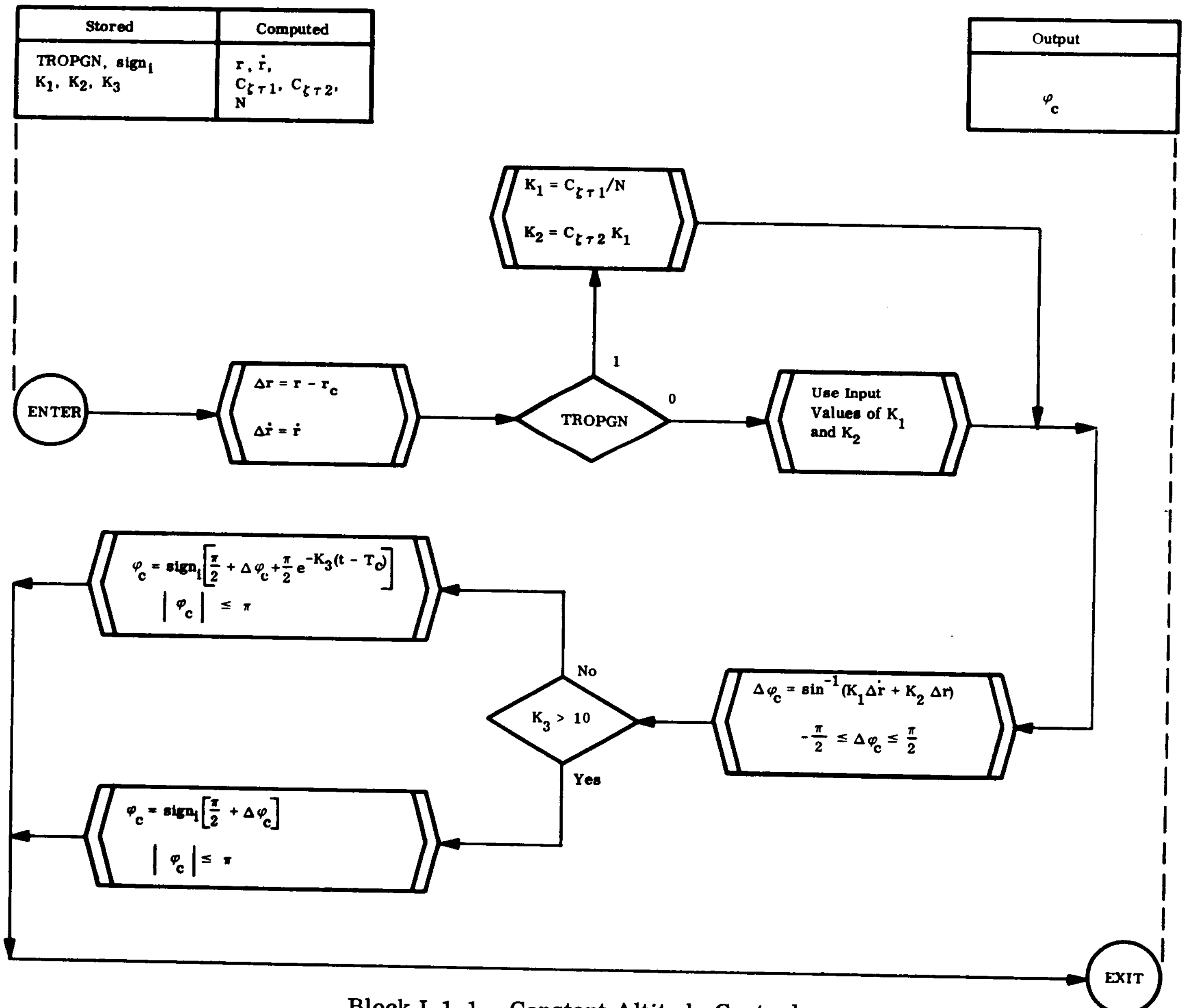


3.4.1.1 Level II Flow Chart - Nominal Trajectory Block

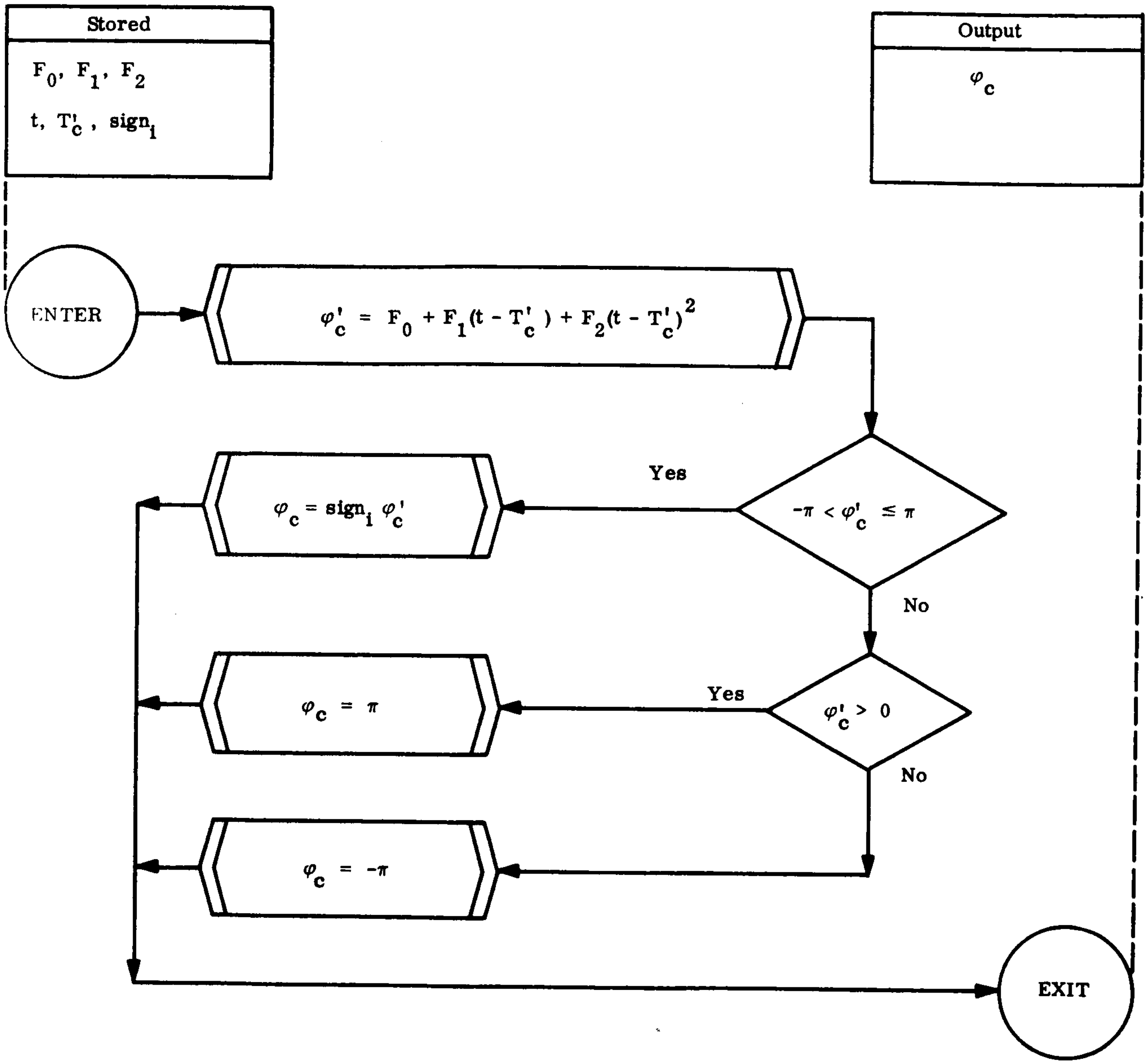
3.4.1.2 Detailed Flow Charts and Equations



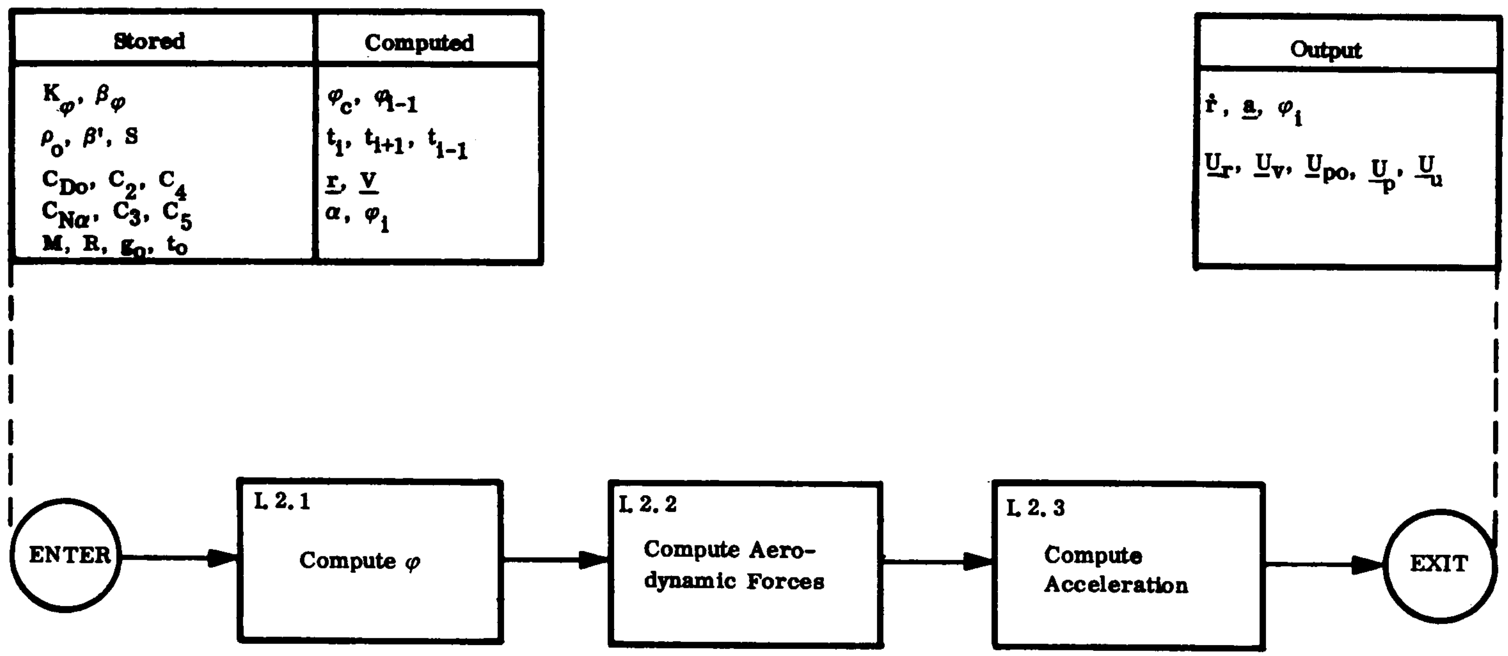
3.4.1.2.1 Nominal Control - Block I.1



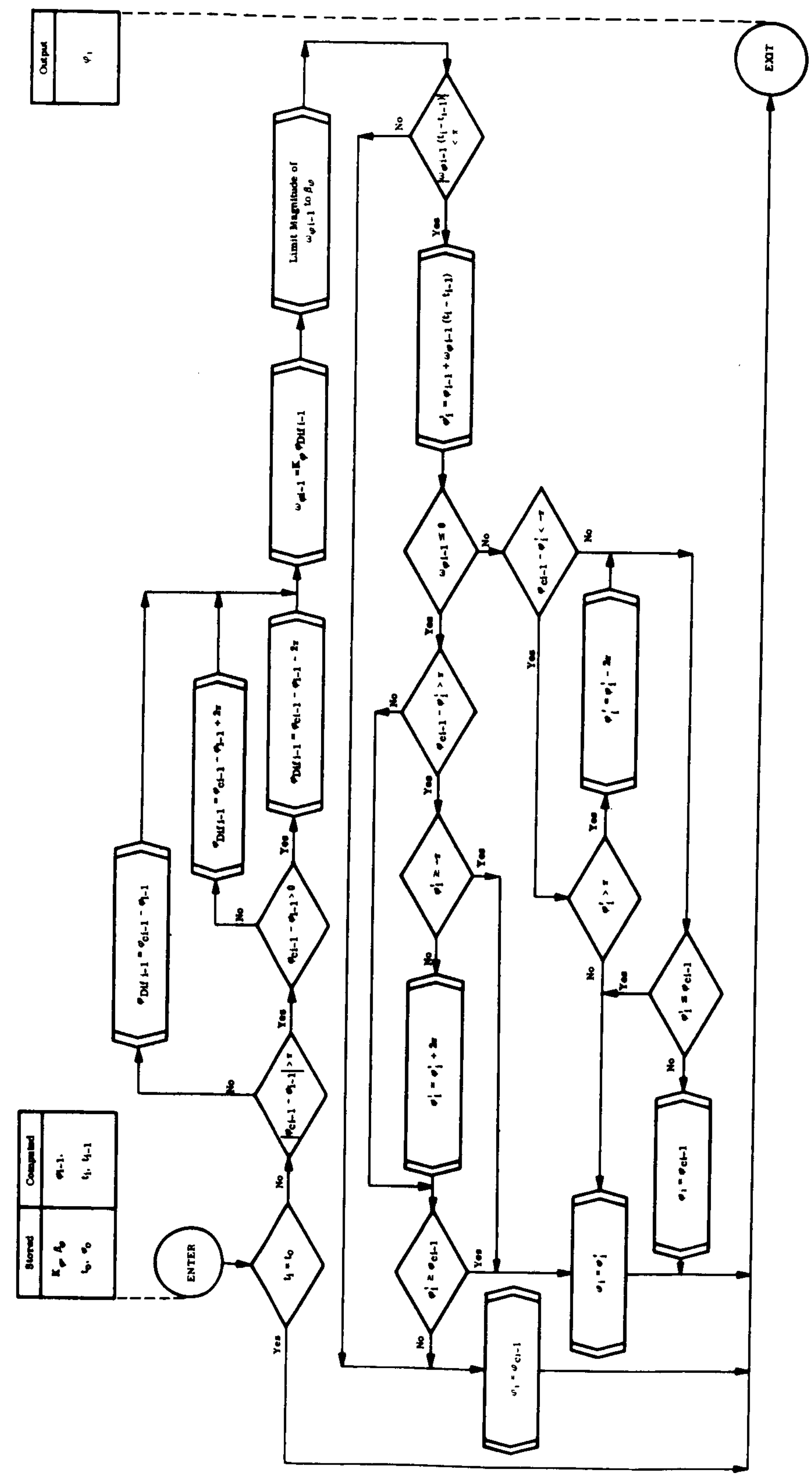
Block I. 1. 1 - Constant Altitude Control



Block I.1.2 Skipout Control



3. 4. 1. 2. 2 Dynamics - Block I. 2



Block I. 2. 1 - Compute φ

Block I. 2. 2 Compute Aerodynamic Forces

INPUT: $\underline{r}$, $\underline{V}$, ρ_o, β', R, S, C_{Do}, C_2, C_4, $C_{N\alpha}$, C_3, C_5, α, φ_i

OUTPUT: $\underline{U}_v$, $\underline{U}_r$, $\underline{U}_{po}$, $\underline{D}$, $\underline{N}$, $\dot{r}$

1. $$V = +\sqrt{\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2}$$

2. $$\underline{U}_v = \frac{\underline{V}}{V}$$

3. $$r = +\sqrt{X^2 + Y^2 + Z^2}$$

4. $$\underline{U}_r = \frac{\underline{r}}{r}$$

5. $$\gamma = \sin^{-1} [\underline{U}_r \cdot \underline{U}_v]$$

6. $$\dot{r} = V \sin \gamma$$

7. $$\underline{U}_u = \frac{\underline{U}_r - \underline{U}_v \sin \gamma}{\cos \gamma}$$

8. $$\underline{U}_p = \underline{U}_u \times \underline{U}_v$$

9. $$\rho = \rho_o e^{-\beta' (r - R)}$$

10. $$C_D = C_{Do} + C_2 \alpha^2 + C_4 \alpha^4$$

11. $$C_N = C_{N\alpha} + C_3 \alpha^3 + C_5 \alpha^5$$

12. $$\underline{D} = - C_D \rho \frac{V^2 S}{2} \underline{U}_v$$

13. $$\underline{N} = C_N \rho \frac{V^2 S}{2} [\cos \varphi_i \underline{U}_u - \sin \varphi_i \underline{U}_p]$$

Block I. 2. 3 Compute Acceleration

INPUT: $\underline{D}$, $\underline{N}$, M, R, g_o

OUTPUT: a_x, a_y, a_z, $\ddot{X}$, $\ddot{Y}$, $\ddot{Z}$, $\underline{a}$, $\underline{f}$

1. $a_x = (D_x + N_x)/M$

2. $a_y = (D_y + N_y)/M$

3. $a_z = (D_z + N_z)/M$

4. $g = g_o \left(\frac{R}{r}\right)^2$

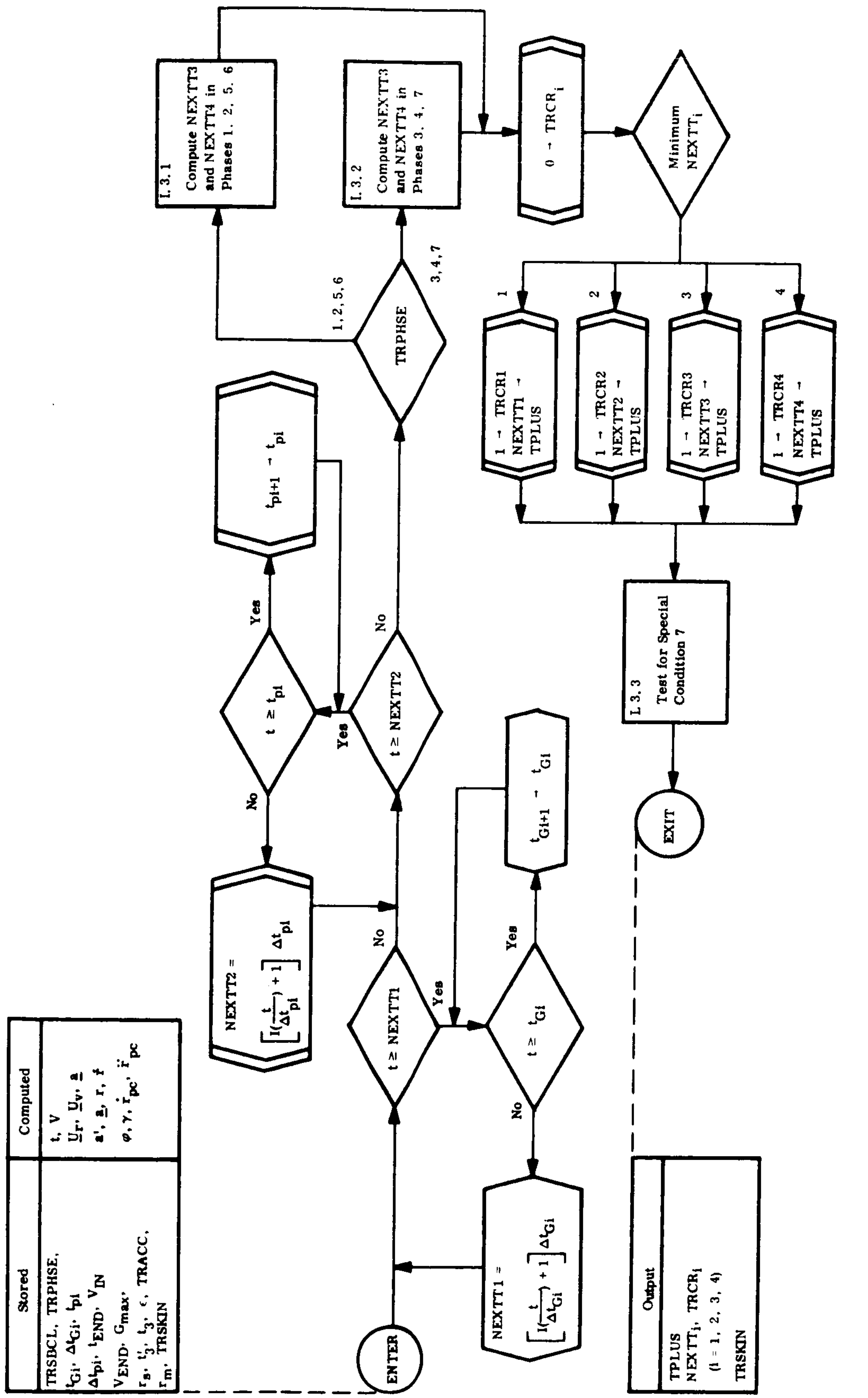
5. $\ddot{X} = a_x - g\frac{X}{r}$

6. $\ddot{Y} = a_y - g\frac{Y}{r}$

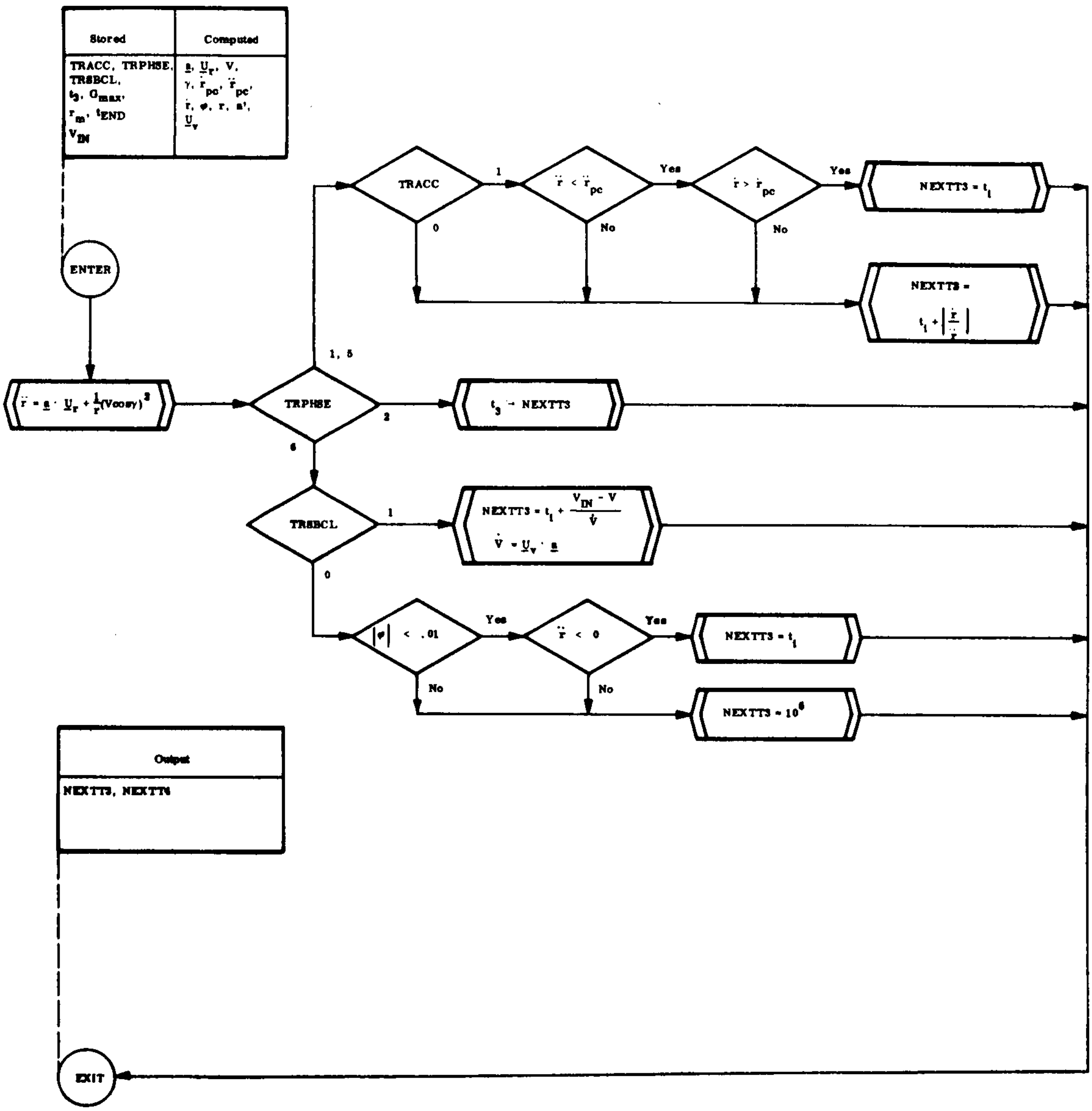
7. $\ddot{Z} = a_z - g\frac{Z}{r}$

8. $\underline{a} \triangleq \ddot{X}\underline{i} + \ddot{Y}\underline{j} + \ddot{Z}\underline{k}$

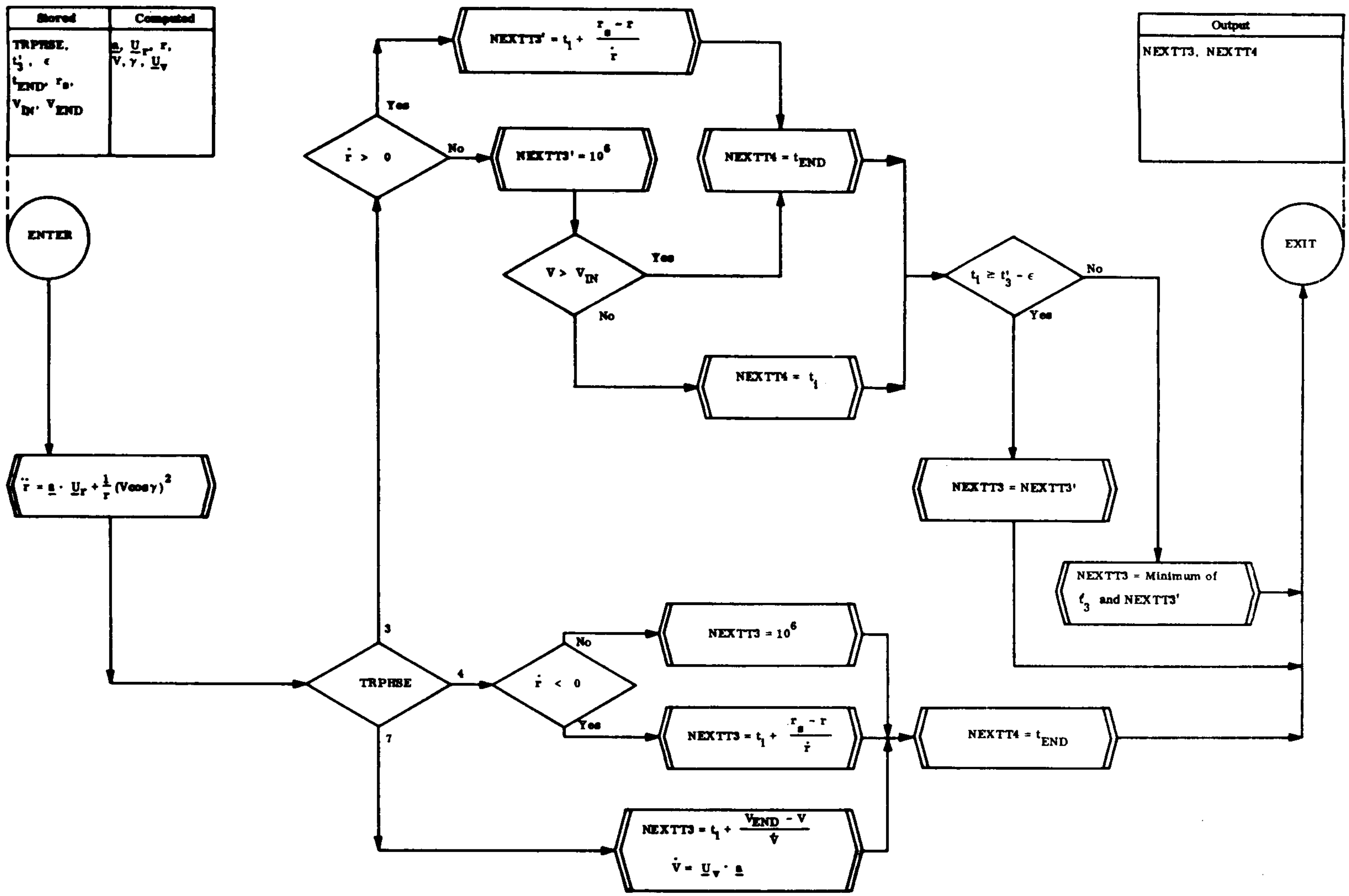
9. $\underline{f} \triangleq a_x\underline{i} + a_y\underline{j} + a_z\underline{k}$



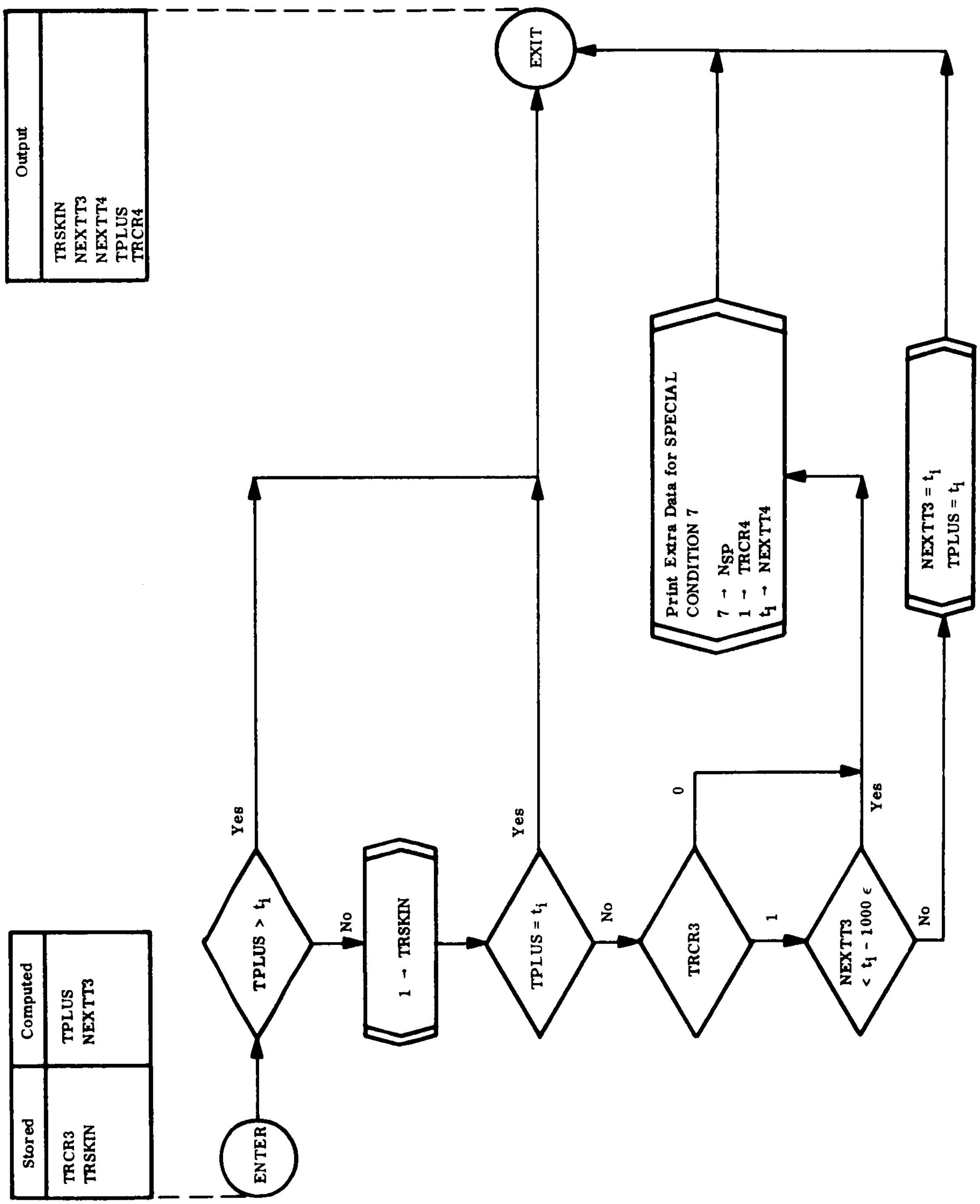
3.4.1.2.3 Compute NEXTTi - Block I.3



Block I. 3. 1 Compute NEXTT3 and NEXTT4 in Phases 1, 2, 5, and 6



Block I. 3. 2 - Compute NEXTT3 and NEXTT4 in Phases 3,4, and 7



Block I.3.3 - Test for Special Condition 7

3.4.1.2.4 Integrate - Block I.4

Input: $\ddot{X}$, $\ddot{Y}$, $\ddot{Z}$, $\underline{f}^*$ (3 x 1), q_s, $\dot{E}_n$, $\dot{\Psi}(t,t_{p-1})$ (6 x 6),

$_c\dot{\Phi}(t,t_{p-1})$, $E_2(t)$ (6 x 2), $E_3(t)\, _c\Phi(t,t_{p-1})$ (6 x 1),

$E_4(t)$ (6 x 2), $\dot{B}(t)$ (6 x 2), $\dot{C}(t)$ (6 x 1), $\dot{\Gamma}(t)$ (6 x 2)

$F_2(t)\ \Phi(t,t_{p-1})$ (6 x 6) {Only the non-zero elements of the above matrices are integrated.}

Output: $\dot{X}$, $\dot{Y}$, $\dot{Z}$, X, Y, Z, $\int_{t_o}^{t} \underline{f}^*\, d\tau$ (3 x 1), Q, E_n

$\Psi(t,t_{p-1})$ (6 x 6), $_c\Phi(t,t_{p-1})$ (1 x 1), $\int_{t_{p-1}}^{t} {_c\Phi}(t,t_{p-1})\, d\tau$ (1 x 1)

$\int_{t_{p-1}}^{t} E_2(\tau)\, d\tau$ (6 x 2), $\int_{t_{p-1}}^{t} E_3(\tau)\ {_c\Phi}(\tau, t_{p-1})\, d\tau$ (6 x 1),

$\int_{t_{p-1}}^{t} E_4(\tau)\, d\tau$ (6 x 2), $\int_{t_{p-1}}^{t} \dot{B}(\tau)\, d\tau$ (6 x 2), $\int_{t_{p-1}}^{t} \dot{C}(\tau)\, d\tau$ (6 x 1),

$\int_{t_{p-1}}^{t} \dot{\Gamma}(\tau)\, d\tau$ (6 x 2), $\int_{t_{p-1}}^{t} F_2(\tau)\ \Phi(\tau, t_{p-1})\, d\tau$ (6 x 6)

$\int_{t_{p-1}}^{t} F_2(t)\ \Phi(t,t_{p-1}) \int_{t_{p-1}}^{t} \dot{B}(\tau)\, d\tau\, dt$ (6 x 2)

$\int_{t_{p-1}}^{t_p} F_2(t)\ \Phi(t,t_{p-1}) \int_{t_{p-1}}^{t} \dot{C}(\tau)\, d\tau\, dt$ (6 x 2)

$\int_{t_{p-1}}^{t} F_2(t)\ \Phi(t,t_{p-1}) \int_{t_{p-1}}^{t} \dot{\Gamma}(\tau)\, d\tau\, dt$ (6 x 2)

The integration routine uses a fixed step size which is input to the program. The input specified above constitutes a partial set of the integrands which, along with the initial conditions, are required to determine the integrals listed in the output. Some of the integrands consist of the output of the integration routine (e.g., $\dot{X}$, $\dot{Y}$, $\dot{Z}$ is output from the integration routine and is used as input to obtain X, Y, Z). The integration equations used are the Gill equations listed below.

1) $$Y_{n+1}^{(1)} = Y_n + \frac{1}{2} \Delta t \, [Y'(t, Y_n)]$$

2) $$Y_{n+1}^{(2)} = Y_{n+1}^{(1)} + \left(\frac{2 - \sqrt{2}}{2}\right) \Delta t \, [Y' (t + \frac{\Delta t}{2}, Y_{n+1}^{(1)}) - Y'(t, Y_n)]$$

3) $$Y_{n+1}^{(3)} = Y_{n+1}^{(2)} + \left(\frac{2 - \sqrt{2}}{2}\right) \Delta t \, [Y' (t + \frac{\Delta t}{2}, Y_{n+1}^{(2)}]$$

$$- \Delta t \, [Y' (t + \frac{\Delta t}{2}, Y_{n+1}^{(1)}] + \left(\frac{1 - \sqrt{2}}{2}\right) \Delta t \, [Y' (t, Y_n)]$$

4) $$Y_{n+1}^{(4)} = Y_{n+1}^{(3)} + \frac{1}{6} \Delta t \, [Y' (t, Y_n) + Y'(t + \Delta t \, Y_{n+1}^{(3)})]$$

$$- \left(\frac{2 + \sqrt{2}}{2}\right) \Delta t \, [Y' (t + \frac{\Delta t}{2}, Y_{n+1}^{(2)})]$$

$$+ \left(\frac{1 + \sqrt{2}}{2}\right) \Delta t \, [Y' (t + \frac{\Delta t}{2}, Y_{n+1}^{(1)})]$$

$$Y_{n+1} = Y_{n+1}^{(4)}$$

3.4.1.2.5 Attitude - Block I. 5

INPUT: φ, α, $\underline{U}_v$, $\underline{U}_p$, $\underline{U}_u$, $\underline{P}_{Io}$, $\underline{Y}_{Ao}$, $\underline{R}_{Oo}$, t_i, t_{i-1}

OUTPUT: α_1, α_2, α_3, ω_{PI}, ω_{YA}, ω_{RO}

$$\begin{bmatrix} \underline{P}_I \\ \underline{Y}_A \\ \underline{R}_O \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \underline{U}_v \\ \underline{U}_p \\ \underline{U}_u \end{bmatrix}$$

$$\alpha_1 = \tan^{-1}\left[\frac{\underline{P}_I \cdot \underline{Y}_{Ao}}{\underline{P}_I \cdot \underline{P}_{Io}}\right] \qquad -\pi < \alpha_1 \leq \pi$$

$$\alpha_2 = \sin^{-1}\left[\underline{P}_I \cdot \underline{R}_{Oo}\right] \qquad -\frac{\pi}{2} \leq \alpha_2 \leq \frac{\pi}{2}$$

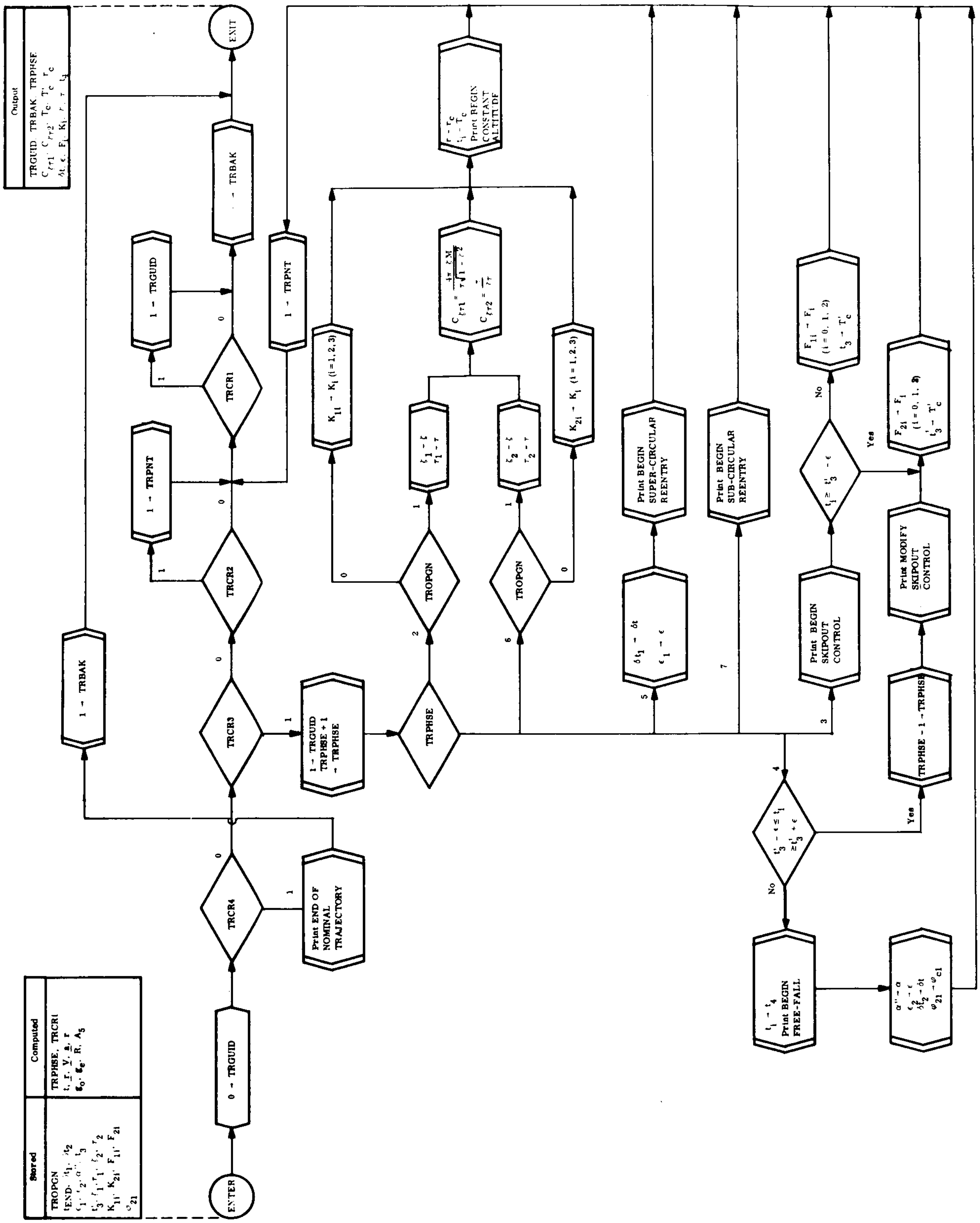
$$\alpha_3 = \tan^{-1}\left[\frac{\underline{Y}_A \cdot \underline{R}_{Oo}}{\underline{R}_O \cdot \underline{R}_{Oo}}\right] \qquad -\pi < \alpha_3 \leq \pi$$

$$\left.\begin{aligned} \dot{\alpha}_1 &= \frac{\alpha_{1i} - \alpha_{1(i-1)}}{t_i - t_{i-1}} \\ \dot{\alpha}_2 &= \frac{\alpha_{2i} - \alpha_{2(i-1)}}{t_i - t_{i-1}} \\ \dot{\alpha}_3 &= \frac{\alpha_{3i} - \alpha_{3(i-1)}}{t_i - t_{i-1}} \end{aligned}\right\} \qquad \dot{\alpha}_1 = \dot{\alpha}_2 = \dot{\alpha}_3 = 0 \qquad \text{at } t = t_o$$

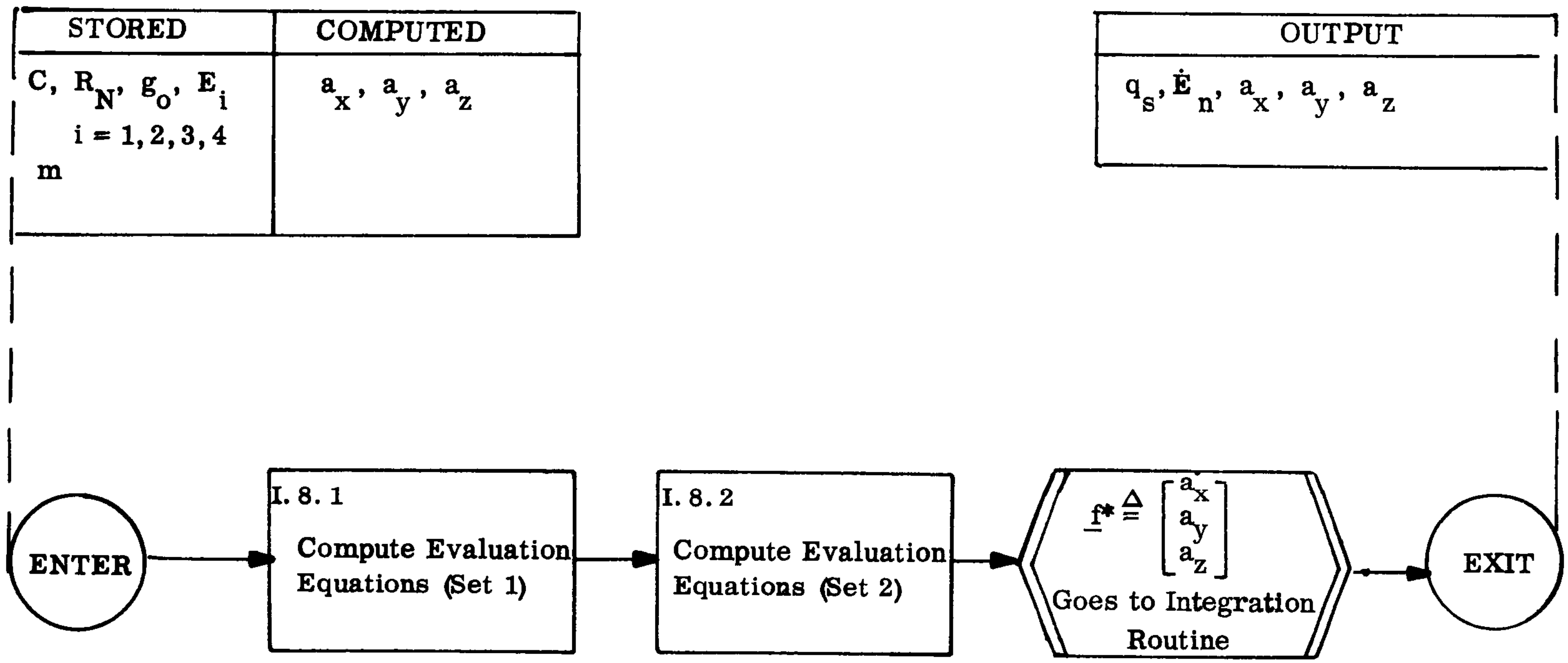
$$\omega_{RO} = \cos\alpha_2 \cos\alpha_3(\dot{\alpha}_1) - \sin\alpha_3(\dot{\alpha}_2)$$

$$\omega_{YA} = \cos\alpha_2 \sin\alpha_3(\dot{\alpha}_1) + \cos\alpha_3(\dot{\alpha}_2)$$

$$\omega_{PI} = -\sin\alpha_2(\dot{\alpha}_1) + \dot{\alpha}_3$$



Block I.6 - Outcontrol



3.4.1.2.7 Evaluation - Block I.8

Block I.8.1 Compute Evaluation Equations (Set 1)

INPUT: C_H, R_N, ρ, ρ_o, g, g_e, C_{e1}, C_{e2}, q_1, q_2, k_H, p_H, V, r, E_i

OUTPUT: q_s, E_n, a'

1. $$q_c = \frac{C_H}{\sqrt{R_N}} \left(\frac{\rho}{\rho_o}\right)^n \left(\frac{V}{\sqrt{g\,r}}\right)^m$$

2. If $\frac{V}{\sqrt{g\,r}} < 1.73$: $q_1 \rightarrow q$; $C_{e1} \rightarrow C_e$

   If $\frac{V}{\sqrt{g\,r}} \geq 1.73$: $q_2 \rightarrow q$; $C_{e2} \rightarrow C_e$

3. $$q_r = k_H R_N \left(\frac{\rho}{\rho_o}\right)^{p_H} C_e V^q$$

4. $$q_s = q_c + q_r$$

5. $$a' = \frac{\sqrt{a_x^2 + a_y^2 + a_z^2}}{g_e}$$

6. $$\tau' = E_o + E_1(a') + E_2(a')^2 + E_3(a')^3 + E_4(a')^4$$

7. $$\dot{E}'_n = \frac{1}{\tau'}$$

8. Is $\dot{E}'_n \leq 0.0008$?

   a. Yes: $\dot{E}_n = 0$

   b. No: $\dot{E}_n = \dot{E}'_n$

Block I. 8. 2 Compute Evaluation Equations (Set 2)

INPUT: $X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}, f, r, g_e, R, A_5$ (3 x 3)

OUTPUT: $\theta, \phi, \beta, a', h$

1. $\underline{r}_t = A_5 \underline{r}$

2. $\underline{V}_t = A_5 \underline{V}$

3. $\theta = \cos^{-1}\left[\frac{Z_t}{r}\right]$

   a. If $Y_t \geq 0$ then $0 \leq \theta \leq \pi$

   b. If $Y_t < 0$ then $\pi < \theta < 2\pi$

4. $\phi' = \tan^{-1}\left[\frac{X_t}{Y_t}\right]$ $\quad -\pi < \phi \leq \pi$

   a. If $\frac{\sqrt{X_t^2 + Y_t^2}}{r} < 0.015$ then $\phi_i = \phi_{i-1}$

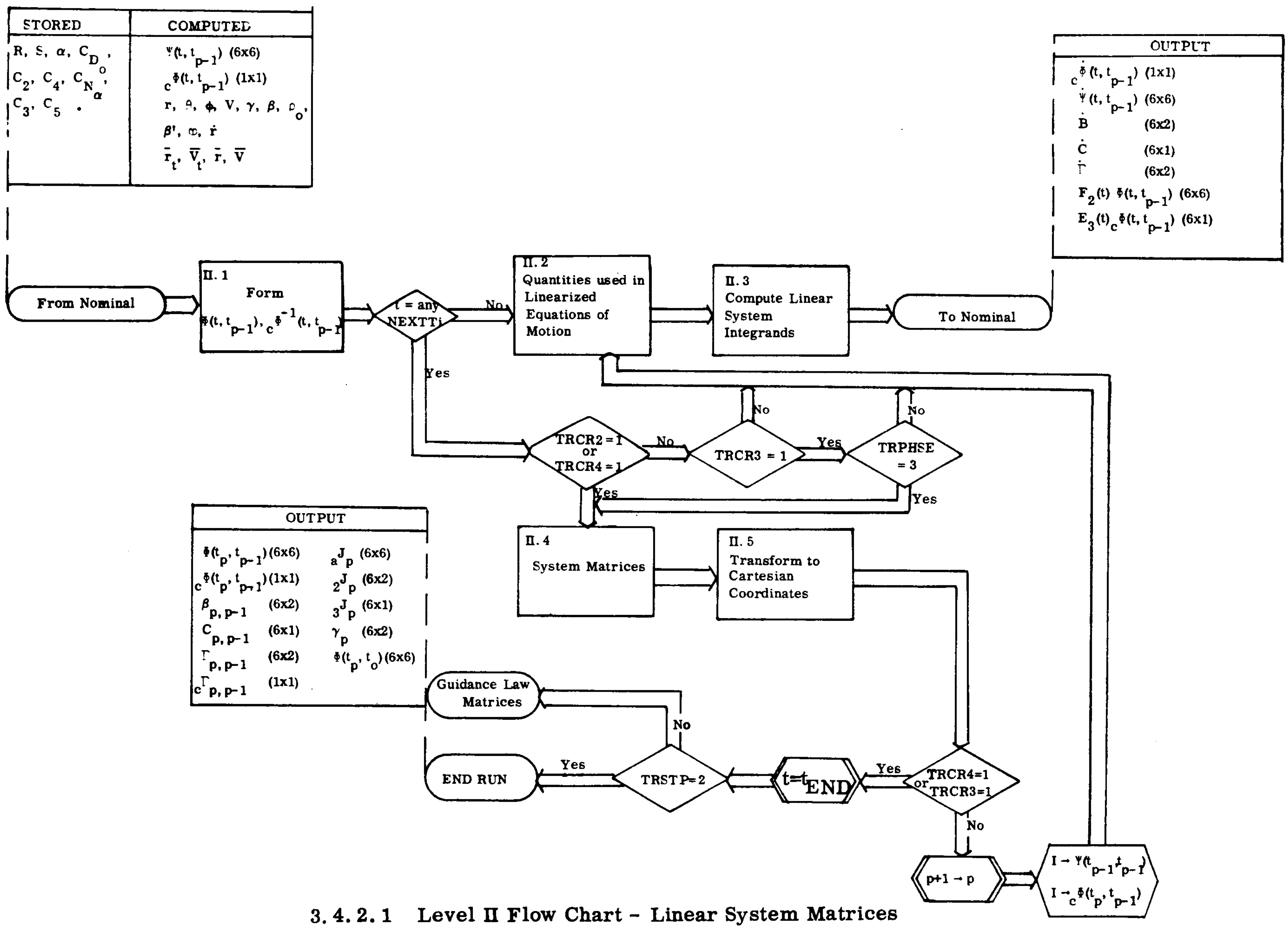
   b. If $\frac{\sqrt{X_t^2 + Y_t^2}}{r} \geq 0.015$ then $\phi_i = \phi'_i$

5. $\beta = \tan^{-1}\left[\frac{\cos\phi \dot{X}_t - \sin\phi \dot{Y}_t}{\cos\theta \sin\phi \dot{X}_t + \cos\theta \cos\phi \dot{Y}_t - \sin\theta \dot{Z}_t}\right]$ $\quad -\pi < \beta < \pi$

6. $a' = \frac{f}{g_e}$

7. $h = (r - R)$

3.4.2 Linear System Matrices - Block II



3.4.2.1 Level II Flow Chart - Linear System Matrices

3.4.2.2 Detailed Flow Charts and Equations

3.4.2.2.1 Form $\Phi(t, t_{p-1})^s$, $\Phi_c^{-1}(t, t_{p-1})^s$ Block II. 1

Input: $\Psi(t, t_{p-1})^s (6x6)$, $_c\Phi(t, t_{p-1})^s (1x1)$

Output: $\Phi(t, t_{p-1})^s (6x6)$, $_c\Phi^{-1}(t, t_{p-1})^s (1x1)$, $\Phi^{-1}(t, t_{p-1})^s (6x6)$, $_c\Phi(t, t_{p-1})^s (6x6)$

1. $\Phi^{-1}(t, t_{p-1})^s \triangleq \Psi^T(t, t_{p-1})^s$

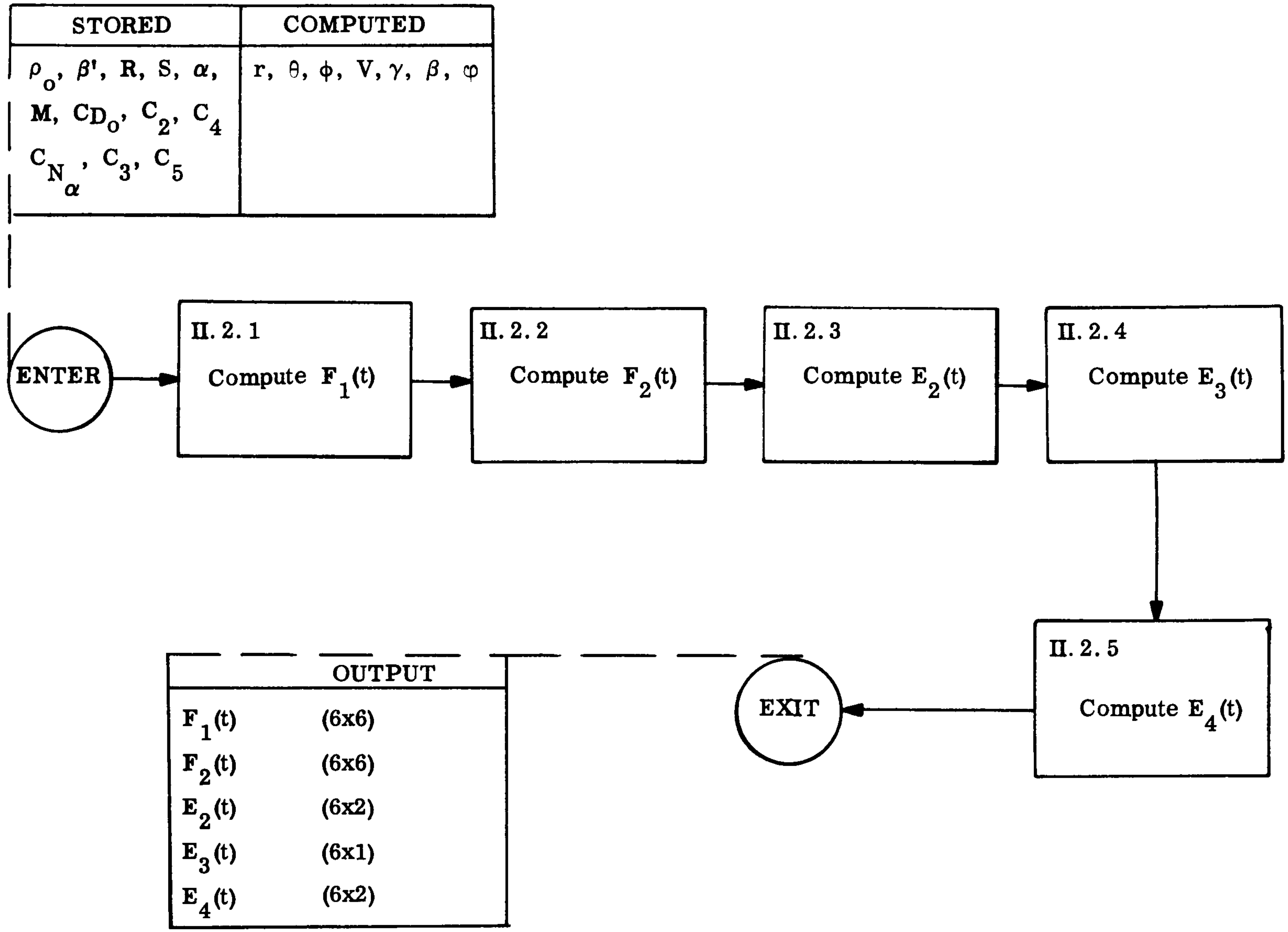
2. $\Phi(t, t_{p-1})^s = [\Phi^{-1}(t, t_{p-1})^s]^{-1}$

3. $_c\Phi^{-1}(t, t_{p-1})^s = [_c\Phi(t, t_{p-1})^s]^{-1}$

4. $\Psi(t, t_{p-1})^s \triangleq \begin{bmatrix} {}_{I}\Psi^s & {}_{II}\Psi^s \\ {}_{III}\Psi^s & {}_{IV}\Psi^s \end{bmatrix}$

5. $\Phi(t, t_{p-1})^s \triangleq \begin{bmatrix} {}_{I}\Phi^s & {}_{II}\Phi^s \\ {}_{III}\Phi^s & {}_{IV}\Phi^s \end{bmatrix}$

Note: In equations 4 and 5 the submatrices are of dimension 3x3.



3.4.2.2.2 Quantities used in Linearized Equations of Motion - Block II. 2

Block II. 2. 1 Compute $F_1(t)$

Input: $r, \theta, \phi, V, \gamma, \beta, \rho_o, \beta', R, S, \alpha, C_{D_o}, C_2, C_4, C_{N_\alpha}, C_3, C_5$

Output: $F_1(t)$ (6x6), D, N

1. $\dot{\theta} = \frac{V}{r} \cos\gamma \cos\beta$

2. $\dot{\phi} = \frac{V \cos\gamma \sin\beta}{r \sin\theta}$ $\qquad \dot{\phi}_i = \dot{\phi}_{i-1}$ if $\sin\theta < 0.015$

   $\cot\theta = 60$ if $\sin\theta < 0.015$

   $\sin\theta = 0.015$ if $\theta < 0.015$

3. $D = C_D \rho V^2 S/2$

4. $N = C_N \rho V^2 S/2$

5. $F_1(t) \triangleq \begin{bmatrix} {}_{I}F_1 & {}_{II}F_1 \\ {}_{III}F_1 & {}_{IV}F_1 \end{bmatrix}$ $\qquad {}_{i}F_1$ are 3x3 matrices, $i = I, II, III, IV$

6. ${}_{I}F_1$:

$${}_{I}f_{1\text{-}11} = {}_{I}f_{1\text{-}12} = {}_{I}f_{1\text{-}13} = 0$$

$${}_{I}f_{1\text{-}21} = \dot{\theta}/r$$

$${}_{I}f_{1\text{-}22} = {}_{I}f_{1\text{-}23} = 0$$

$${}_{I}f_{1\text{-}31} = -\dot{\phi}/r$$

$${}_{I}f_{1\text{-}32} = -\dot{\phi}\cot\theta$$

$${}_{I}f_{1\text{-}33} = 0$$

7. $_{II}F_1$:

$$_{II}f_{1-11} = \sin\gamma$$

$$_{II}f_{1-12} = V\cos\gamma$$

$$_{II}f_{1-13} = 0$$

$$_{II}f_{1-21} = \cos\gamma\cos\beta/r = \dot{\theta}/V$$

$$_{II}f_{1-22} = -V/r\sin\gamma\cos\beta = -\tan\gamma\,\dot{\theta}$$

$$_{II}f_{1-23} = -\frac{V}{r}\cos\gamma\sin\beta = -\dot{\phi}\sin\theta$$

$$_{II}f_{1-31} = \frac{\cos\gamma\sin\beta}{r\sin\theta} = \dot{\phi}/V$$

$$_{II}f_{1-32} = \frac{-V\sin\gamma\sin\beta}{r\sin\theta} = -\dot{\phi}\tan\gamma$$

$$_{II}f_{1-33} = \frac{\dot{\theta}}{\sin\theta}$$

8. $_{III}F_1$:

$$_{III}f_{1-11} = \frac{2g\sin\gamma}{r}$$

$$_{III}f_{1-12} = {}_{III}f_{1-13} = 0$$

$$_{III}f_{1-21} = \frac{2g\cos\gamma}{rV} - \frac{V}{r^2}\cos\gamma$$

$$_{III}f_{1-22} = {}_{III}f_{1-23} = 0$$

$$_{III}f_{1-31} = -\frac{\dot{\phi}\cos\theta}{r}$$

$$_{III}f_{1-32} = -\dot{\phi}/\sin\theta$$

$$_{III}f_{1-33} = 0$$

9. ${}_{IV}F_1$:

$${}_{IV}f_{1-11} = {}_{IV}f_{1-13} = 0$$

$${}_{IV}f_{1-12} = - g \cos \gamma$$

$${}_{IV}f_{1-21} = \frac{g}{V^2} \cos \gamma + \frac{\cos \gamma}{r}$$

$${}_{IV}f_{1-22} = - (\frac{V}{r} - \frac{g}{V}) \sin \gamma$$

$${}_{IV}f_{1-23} = 0$$

$${}_{IV}f_{1-31} = \frac{\dot{\phi} \cos \theta}{V}$$

$${}_{IV}f_{1-32} = - \dot{\phi} \tan \gamma \cos \theta$$

$${}_{IV}f_{1-33} = \dot{\theta} \cot \theta$$

Block II. 2. 2 Compute $F_2(t)$

Input: β', ρ, D, N, M, φ, V

Output: $F_2(t)$ (6x6)

1. $F_2(t) \triangleq \begin{bmatrix} {}_{I}F_2 & {}_{II}F_2 \\ {}_{III}F_2 & {}_{IV}F_2 \end{bmatrix}$ ${}_{i}F_2$ are 3x3 matrices, i = I, II, III, IV

2. ${}_{I}F_2 = {}_{II}F_2 = 0$

3. ${}_{III}F_2$:

$${}_{III}f_{2-11} = \frac{\beta' D}{M}$$

$${}_{III}f_{2-12} = {}_{III}f_{2-13} = 0$$

$${}_{III}f_{2-21} = -\beta' \frac{N \cos \varphi}{MV}$$

$${}_{III}f_{2-22} = {}_{III}f_{2-23} = 0$$

$${}_{III}f_{2-31} = -\frac{\sin \varphi \, \beta' N}{MV}$$

$${}_{III}f_{2-32} = {}_{III}f_{2-33} = 0$$

4. ${}_{IV}F_2$:

$${}_{IV}f_{2-11} = -\frac{D}{M} \frac{2}{V}$$

$${}_{IV}f_{2-12} = {}_{IV}f_{2-13} = 0$$

$${}_{IV}f_{2-21} = \frac{N \cos \varphi}{MV^2}$$

$${}_{IV}f_{2-22} = {}_{IV}f_{2-23} = 0$$

$${}_{IV}f_{2-31} = \frac{N \sin \varphi}{MV^2} \qquad {}_{IV}f_{2-32} = {}_{IV}f_{2-33} = 0$$

Block II. 2. 3 Compute $E_2(t)$

Input: D, M, C_{D_o}, C_2, C_4, φ, ρ, S, V, α

Output: $E_2(t)$ (6x2)

1. $$E_2(t) = \begin{bmatrix} {}_{I}E_2 \\ {}_{II}E_2 \end{bmatrix}$$ ${}_{i}E_2$ are 3x2 matrices, $i = I, II$

2. $${}_{I}E_2 = 0$$

3. ${}_{II}E_2$:

$${}_{II}e_{2-11} = -\frac{D}{M} \cdot \frac{1}{(C_{D_o} + C_2\alpha^2 + C_4\alpha^4)}$$

$${}_{II}e_{2-12} = 0$$

$${}_{II}e_{2-21} = 0$$

$${}_{II}e_{2-22} = \frac{\alpha \cos\varphi}{2M} \rho S V$$

$${}_{II}e_{2-31} = 0$$

$${}_{II}e_{2-32} = \frac{\alpha \sin\varphi}{2M} \rho S V$$

Block II. 2.4 Compute $E_3(t)$

Input: D, N, φ, ρ_o, V, M

Output: $E_3(t)$ (6x1)

$$E_3(t) = \begin{bmatrix} {}_{I}E_3 \\ {}_{II}E_3 \end{bmatrix}$$

${}_{i}E_3$ are 3x1 matrices

i = I, II

$${}_{I}E_3 = 0$$

${}_{II}E_3$:

$${}_{II}e_{3-11} = -\frac{D}{M}\,\frac{1}{\rho_o}$$

$${}_{II}e_{3-21} = \frac{N}{M}\,\frac{\cos\varphi}{V}\,\frac{1}{\rho_o}$$

$${}_{II}e_{3-31} = \frac{N}{M}\,\frac{\sin\varphi}{V}\,\frac{1}{\rho_o}$$

Block II. 2. 5 Compute $E_4(t)$

Input: M, C_{N_α}, C_3, C_5, α, ρ, V, S. φ, N

Output: $E_4(t)$ (6x2)

1. $$E_4(t) = \begin{bmatrix} {}_{I}E_4 \\ {}_{II}E_4 \end{bmatrix} \qquad {}_{i}E_4 = \text{3x2 matrices} \qquad i = I, II$$

2. ${}_{I}E_4 = 0$

3. ${}_{II}E_4$

$$ {}_{II}e_{4-11} = -\frac{1}{M}(C_2\alpha + 2C_4\alpha^3)\rho V^2 S$$

$$ {}_{II}e_{4-12} = 0$$

$$ {}_{II}e_{4-21} = \frac{\cos\varphi}{2M}(C_{N_\alpha} + 3C_3\alpha^2 + 5C_5\alpha^4)\rho V S$$

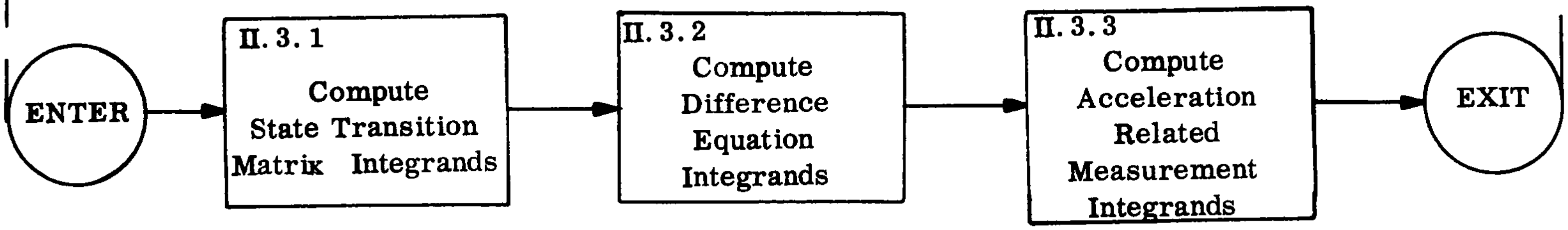
$$ {}_{II}e_{4-22} = \frac{-N\sin\varphi}{MV}$$

$$ {}_{II}e_{4-31} = \frac{\sin\varphi}{2M}(C_{N_\alpha} + 3C_3\alpha^2 + 5C_5\alpha^4)\rho V S$$

$$ {}_{II}e_{4-32} = \frac{N\cos\varphi}{MV}$$

STORED	COMPUTED
	$\dot{r}$, h
	$F_1(t)$ (6x6), $E_2(t)$ (6x2)
	$F_2(t)$ (6x6), $E_3(t)$ (6x1)
	$\Psi(t, t_{p-1})^S$(6x6), $E_4(t)$ (6x2)
	${}_c\Phi(t, t_{p-1})^S$(1x1),
	$\Phi(t, t_{p-1})^S$ (6x6)

OUTPUT	
${}_c\Phi(t, t_{p-1})^S$(1x1),	$F_2(t)\ \Phi(t, t_{p-1})^S$ (6x6)
$\Psi(t, t_{p-1})^S$ (6x6),	$E_3(t)\ {}_c\Phi(t, t_{p-1})$ (6x1)
$\dot{B}^S$(6x2)	
$\dot{C}^S$(6x1)	
$\dot{\Gamma}^S$ (6x2)	



3. 4. 2. 2. 3 Compute Linear Systems Integrands - Block II. 3

Block II. 3. 1 Compute State Transition Matrix

Input: $\dot{r}$, h, ${}_c\Phi(t, t_{p-1})^S (1x1)$, $F_1(t)(6x6)$, $F_2(t)(6x6)$, $\Psi(t, t_{p-1})^S (6x6)$, h_ρ

Output: ${}_c\dot{\Phi}(t, t_{p-1})^S (1x1)$, $\dot{\Psi}(t, t_{p-1})^S (6x6)$

1. $${}_c\dot{\Phi}(t, t_{p-1})^S = - \frac{|\dot{r}|}{h_\rho} {}_c\Phi(t, t_{p-1})^S$$

2. $${}_c\Phi(t_{p-1}, t_{p-1})^S = I$$

3. $$\dot{\Psi}(t, t_{p-1})^S = - [F_1(t) + F_2(t)]^T \Psi(t, t_{p-1})^S$$

4. $$\Psi(t_{p-1}, t_{p-1})^S = I$$

Note that the partitioned matrices ${}_I F_2$ and ${}_{II}F_2$ (defined in Block II. 2. 2) are composed of zeros and that 6 of the 9 elements of ${}_I F$ (defined in II. 2. 1) are zero.

Block II. 3. 2 Compute Difference Equation Integrands

Input: $E_2(t)$ (6x2), $E_3(t)$ (6x1), $E_4(t)$ (6x2), $\Psi(t, t_{p-1})^S$ (6x6), ${}_c\Phi(t, t_{p-1})^S$ (1x1)

Output: $\dot{B}^S$ (6x2), $\dot{C}^S$ (6x1), $\dot{\Gamma}^S$ (6x2)

1. $\dot{B}^S = \Phi^{-1}(t, t_{p-1})^S \, E_2(t)$
2. $\dot{C}^S = \Phi^{-1}(t, t_{p-1})^S \, E_3(t) \, {}_c\Phi(t, t_{p-1})^S$
3. $\dot{\Gamma}^S = \Phi^{-1}(t, t_{p-1})^S \, E_4(t)$

Block II. 3. 3 Compute Acceleration Related Measurement Integrands

Input: $F_2(t)(6x6)$, $\overset{S}{\Phi(t, t_{p-1})}(6x6)$, $E_3(t)(6x1)$, $\overset{S}{{}_c\Phi(t, t_{p-1})}(1x1)$

Output: $F_2(t)\ \overset{S}{\Phi(t, t_{p-1})}(6x6)$, $E_3(t)\ \overset{S}{{}_c\Phi(t, t_{p-1})}(6x1)$

1. $$F_2(t)\ \Phi(t, t_{p-1})^S = \begin{bmatrix} 0 & 0 \\ [({}_{III}F_2)({}_I\Phi)^S + ({}_{IV}F_2)({}_{III}\Phi)^S] & [({}_{III}F_2)({}_{II}\Phi)^S + ({}_{IV}F_2)({}_{IV}\Phi)^S] \end{bmatrix}$$

2. $$E_3(t)\ {}_c\Phi(t, t_{p-1})^S = \begin{bmatrix} 0 \\ {}_{II}E_3 \end{bmatrix} {}_c\Phi(t, t_{p-1})^S$$

Notice that the top 3 rows of both of these matrices are empty (i. e., elements equal zero). They need not be computed and should not be sent to the integration routine.

STORED	COMPUTED	OUTPUT
	$\Phi(t_p, t_{p-1})^S$ (6x6)	$B^S_{p,p-1}$ (6x2), $\quad {}_aJ^S_p$ (6x6)
	${}_c\Phi(t_p, t_{p-1})^S$ (1x1)	$C^S_{p,p-1}$ (6x1), $\quad {}_2J^S_p$ (6x2)
	$\int_{t_{p-1}}^{t_p} \dot{B}^S \, d\tau$ (6x2)	$\Gamma^S_{p,p-1}$ (6x2) $\quad {}_3J^S_p$ (6x1)
	$\int_{t_{p-1}}^{t_p} \dot{C}^S \, d\tau$ (6x1)	${}_c\Gamma^S_{p,p-1}$ (1x1) $\quad \gamma^S_p$ (6x2)
	$\int_{t_{p-1}}^{t_p} \dot{\Gamma}^S \, d\tau$ (6x2)	$\Phi(t_p, t_o)^S$ (6x6)
	$\dot{r}$	

ENTER → II. 4. 1 Compute Difference Equation Coefficients → II. 4. 2 Compute Acceleration Related Measurements Coefficients → II. 4. 3 Compute $\Phi(t_p, t_o)$ → EXIT

3. 4. 2. 2. 4 System Matrices - Block II. 4

Block II. 4. 1 Compute Difference Equation Coefficients

Input: $\Phi(t_p, t_{p-1})^s (6x6)$, $\int_{t_{p-1}}^{t_p} \dot{B}^s \, d\tau$ (6x2)

$\int_{t_{p-1}}^{t_{p-1}} \dot{C}^s \, d\tau$ (6x1)

$\int_{t_{p-1}}^{t_p} \dot{\Gamma}^s \, d\tau$ (6x2)

${}_c\Phi(t_p, t_{p-1})^s (1x1)$, $\int_{t_{p-1}}^{t_p} {}_c\Phi^{-1}(\tau, t_{p-1})^s \, d\tau$

Output: $B^s_{p,p-1}$ (6x2), $C^s_{p,p-1}$ (6x1), $\Gamma^s_{p,p-1}$ (6x2), ${}_c\Gamma^s_{p,p-1}$ (1x1)

1. $$B^s_{p,p-1} = \Phi(t_p, t_{p-1})^s \int_{t_{p-1}}^{t_p} \dot{B}^s \, d\tau$$

2. $$C^s_{p,p-1} = \Phi(t_p, t_{p-1})^s \int_{t_{p-1}}^{t_p} \dot{C}^s \, d\tau$$

3. $$\Gamma^s_{p,p-1} = \Phi(t_p, t_{p-1})^s \int_{t_{p-1}}^{t_p} \dot{\Gamma}^s \, d\tau$$

4. $${}_c\Gamma^s_{p,p-1} = {}_c\Phi(t_p, t_{p-1})^s \int_{t_{p-1}}^{t_p} {}_c\Phi^{-1}(\tau, t_{p-1})^s \, d\tau$$

Block II. 4. 2 Compute Acceleration Related Measurement Coefficients

Input: ${}_a J^s_{p-1}$ (6x6), $\Phi^{-1}(t_p, t_{p-1})^s$ (6x6), ${}_c\Phi^{-1}(t_p, t_{p-1})^s$ (1x1), $\int_{t_{p-1}}^{t_p} \dot{B}(t)^s dt$ (6x2)

$\int_{t_{p-1}}^{t_p} F_2(t)\, \Phi(t, t_{p-1})^s dt$ (6x6), ${}_2 J^s_{p-1}$ (6x2), $B(t_p, t_{p-1})^s$ (6x2), $\int_{t_{p-1}}^{t_p} \dot{C}(t)^s dt$ (6x1)

$\int_{t_{p-1}}^{t_p} F_2(t)\, \Phi(t, t_{p-1})^s \int_{t_{p-1}}^{t} \dot{B}(t)^s\, d\tau dt$ (6x2), ${}_3 J^s_{p-1}$ (6x1), $C(t_p, t_{p-1})^s$ (6x1),

$\int_{t_{p-1}}^{t_p} E_3(t)\, {}_c\Phi(t, t_{p-1})^s dt$ (6x1), $\int_{t_{p-1}}^{t_p} F_2(t)\, \Phi(t, t_{p-1})^s \int_{t_{p-1}}^{t} \dot{C}(\tau)^s d\tau dt$ (6x1),

$\int_{t_{p-1}}^{t_p} \dot{\Gamma}(t)^s dt$ (6x2)

$\Gamma(t_p, t_{p-1})^s$ (6x2), $\int_{t_{p-1}}^{t_p} E_4(t)\, dt$ (6x2), $\int_{t_{p-1}}^{t_p} F_2(t)\, \Phi(t, t_{p-1})^s \int_{t_{p-1}}^{t} \dot{\Gamma}(\tau)^s d\tau dt$ (6x2),

$\int_{t_{p-1}}^{t_p} E_2(t) dt$

Output: ${}_a J^s_p$ (6x6), ${}_2 J^s_p$ (6x2), ${}_3 J^s_p$ (6x1), γ^s_p (6x2)

Note: The top three rows of the following matrices are empty (i. e., the elements are equal to zero).

1. $${}_a J^s_p = {}_a J_{p-1}\, \Phi^{-1}(t_p, t_{p-1})^s + \int_{t_{p-1}}^{t} F_2(t)\, \Phi(t, t_{p-1})^s dt\, \Phi^{-1}(t_p, t_{p-1})^s$$

2. $${}_2 J^s_p = {}_2 J^s_{p-1} - {}_a J^s_{p-1} \int_{t_{p-1}}^{t_p} \dot{B}\, dt + \int_{t_{p-1}}^{t_p} E_2(t)\, dt - \int_{t_{p-1}}^{t_p} F_2(t)\, \Phi(t, t_{p-1})^s dt \int_{t_{p-1}}^{t_p} \dot{B}(t)^s dt$$
$$+ \int_{t_{p-1}}^{t_p} F_2(t)\, \Phi(t, t_{p-1})^s \int_{t_{p-1}}^{t} \dot{B}(\tau)^s d\tau\, dt$$

3. $${}_3J_p^s = {}_3J_{p-1}^s - {}_aJ_{p-1}^s \int_{t_{p-1}}^{t_p} \dot{C}\, dt \; {}_c\Phi^{-1}(t_p, t_{p-1})^s$$

$$+ \int_{t_{p-1}}^{t_p} E_3(t)\; {}_c\Phi(t, t_{p-1})^s dt \; {}_c\Phi^{-1}(t_p, t_{p-1})^s$$

$$- \int_{t_{p-1}}^{t_p} F_2(t)\; \Phi(t, t_{p-1})^s dt \int_{t_{p-1}}^{t_p} \dot{C}(t)^s dt \; {}_c\Phi^{-1}(t_p, t_{p-1})^s$$

$$+ \int_{t_{p-1}}^{t_p} F_2(t)\; \Phi(t, t_{p-1})^s \int_{t_{p-1}}^{t} \dot{C}(\tau)^s d\tau\, dt \; {}_c\Phi^{-1}(t_p, t_{p-1})^s$$

4. $$\gamma_p^s = -{}_aJ_{p-1}^s \int_{t_{p-1}}^{t_p} \dot{\Gamma} dt + \int_{t_{p-1}}^{t_p} E_4(t)\, dt$$

$$- \int_{t_{p-1}}^{t_p} F_2(t)\; \Phi(t, t_{p-1})^s dt \int_{t_{p-1}}^{t_p} \dot{\Gamma}(t)^s dt$$

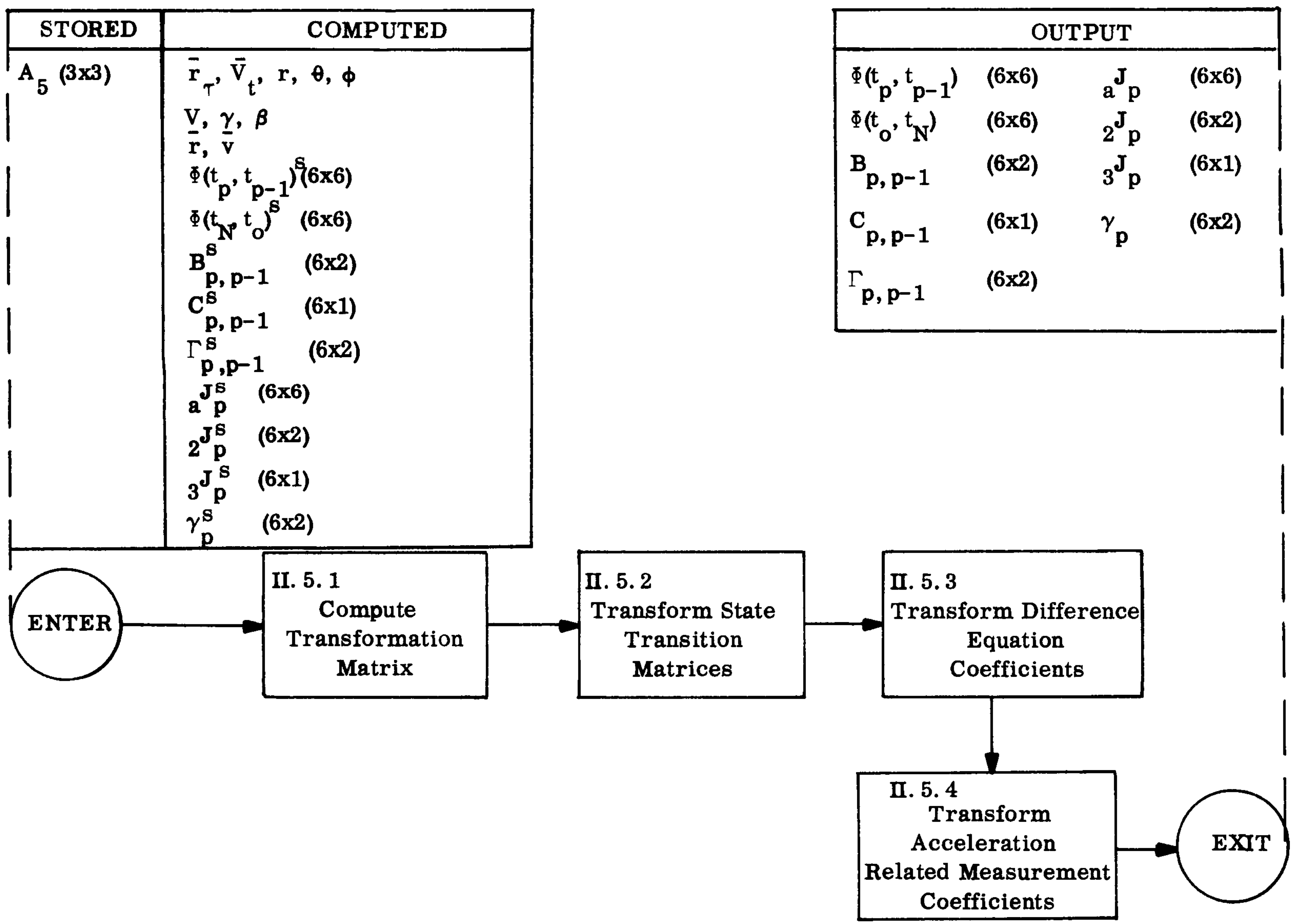
$$+ \int_{t_{p-1}}^{t_p} F_2(t)\; \Phi(t, t_{p-1})^s \int_{t_{p-1}}^{t} \dot{\Gamma}(\tau)^s d\tau\, dt$$

Block II. 4. 3 Compute $\Phi(t_p, t_o)$

Input: $\Phi(t_p, t_{p-1})^S$ (6x6), $\Phi(t_{p-1}, t_o)^S$ (6x6)

Output: $\Phi(t_p, t_o)^S$ (6x6)

1. $\Phi(t_p, t_o)^S = \Phi(t_p, t_{p-1})^S \, \Phi(t_{p-1}, t_o)^S$



3. 4. 2. 2. 5 Transform to Cartesian Coordinates - Block II. 5

Block II. 5. 1 Compute Transformation Matrix

Input: A_5 (3x3), $\bar{r}_t$, $\bar{V}_t$, $\bar{r}$, $\bar{V}$, r, θ, ϕ, V, γ, β

Output: A_6 (6x6), A_6^{-1} (6x6)

(Note: elements not computed are equal to zero.)

1. $$\underset{\text{Spherical}}{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}} \triangleq \begin{bmatrix} a_{6-11} & a_{6-12} & a_{6-13} & \cdots & a_{6-16} \\ & & & & \\ & & & & \\ & & & & \\ & & & & \\ & \cdots & & & a_{6-66} \end{bmatrix} \underset{\text{Cartesian}}{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}}$$

2. $$C_6 = \frac{\cos^2\beta\,(-\sin\phi\,\dot{X}_t - \tan\beta\cos\theta\cos\phi\,\dot{X}_t + \tan\beta\cos\theta\sin\phi\,\dot{Y}_t - \cos\phi\,\dot{Y}_t)}{\cos\theta\sin\phi\,\dot{X}_t + \cos\theta\cos\phi\,\dot{Y}_t - \sin\theta\,\dot{Z}_t}$$

3. $$C_7 = \frac{\sin\beta\cos\beta\quad(\sin\theta\sin\phi\,\dot{X}_t + \sin\theta\cos\phi\,\dot{Y}_t + \cos\theta\,\dot{Z}_t}{\cos\theta\sin\phi\,\dot{X}_t + \cos\theta\cos\phi\,\dot{Y}_t - \sin\theta\,\dot{Z}_t}$$

4. $$C_8 = \frac{\cos^2\beta\quad(\cos\phi - \tan\beta\cos\theta\sin\phi)}{\cos\theta\sin\phi\,\dot{X}_t + \cos\theta\cos\phi\,\dot{Y}_t - \sin\theta\,\dot{Z}_t}$$

5. $$C_9 = \frac{\cos^2\beta\,(-\sin\phi - \tan\beta\cos\theta\cos\phi)}{\cos\theta\sin\phi\,\dot{X}_t + \cos\theta\cos\phi\,\dot{Y}_t - \sin\theta\,\dot{Z}_t}$$

6. $$C_{10} = \frac{\cos^2\beta\sin\theta\tan\beta}{\cos\theta\sin\phi\,\dot{X}_t + \cos\theta\cos\phi\,\dot{Y}_t - \sin\theta\,\dot{Z}_t}$$

7.
$$a_{6-11} = \frac{X}{r} = U_x$$
$$a_{6-12} = \frac{Y}{r} = U_y$$
$$a_{6-13} = \frac{Z}{r} = U_z$$

8.
$$a_{6-21} = \frac{1}{r \sin \theta} \left(\frac{X}{r} \cos \theta - a_{5-31}\right)$$
$$a_{6-22} = \frac{1}{r \sin \theta} \left(\frac{Y}{r} \cos \theta - a_{5-32}\right)$$
$$a_{6-23} = \frac{1}{r \sin \theta} \left(\frac{Z}{r} \cos - a_{5-33}\right)$$

when $\sin \theta < 0.015$ use last values for a_{6-21}, a_{6-22}, a_{6-23}, a_{6-31}, a_{6-32}, a_{6-33}

9.
$$a_{6-31} = \frac{\cos^2 \phi}{Y_t} \left(a_{5-11} - \tan \phi \, a_{5-21}\right)$$
$$a_{6-32} = \frac{\cos^2 \phi}{Y_t} \left(a_{5-12} - \tan \phi \, a_{5-22}\right)$$
$$a_{6-33} = \frac{\cos^2 \phi}{Y_t} \left(a_{5-13} - \tan \phi \, a_{5-23}\right)$$

10.
$$a_{6-44} = \frac{\dot{X}}{V}$$
$$a_{6-45} = \frac{\dot{Y}}{V}$$
$$a_{6-46} = \frac{\dot{Z}}{V}$$

11.
$$a_{6-51} = \frac{\dot{X} - V \sin \gamma \frac{X}{r}}{r V \cos \gamma}$$
$$a_{6-52} = \frac{\dot{Y} - V \sin \gamma \frac{Y}{r}}{r V \cos \gamma}$$

$$a_{6-53} = \frac{\dot{Z} - V \sin\gamma \dfrac{Z}{r}}{r\, V \cos\gamma}$$

$$a_{6-54} = \frac{X - r \sin\gamma \dfrac{\dot{X}}{V}}{r\, V \cos\gamma}$$

$$a_{6-55} = \frac{Y - r \sin\gamma \dfrac{\dot{Y}}{V}}{r\, V \cos\gamma}$$

$$a_{6-56} = \frac{Z - r \sin\gamma \dfrac{\dot{Z}}{V}}{r\, V \cos\gamma}$$

12. $a_{6-61} = C_4\, a_{6-31} + C_5\, a_{6-21}$

$a_{6-62} = C_4\, a_{6-32} + C_5\, a_{6-22}$

$a_{6-63} = C_4\, a_{6-33} + C_5\, a_{6-23}$

$a_{6-64} = C_8\, a_{5-11} + C_9\, a_{5-21} + C_{10}\, a_{5-31}$

$a_{6-65} = C_8\, a_{5-12} + C_9\, a_{5-22} + C_{10}\, a_{5-32}$

$a_{6-66} = C_8\, a_{5-13} + C_9\, a_{5-23} + C_{10}\, a_{5-33}$

13. Compute A_6^{-1}

Note: the upper right 3x3 submatrix consists of zeros.

Block II. 5. 2 Transform State Transition Matrix

Input: $\Phi(t_p, t_{p-1})^S (6x6)$, $A_6(t_p)(6x6)$, $\Phi(t_N, t_o)^S (6x6)$

Output: $\Phi(t_p, t_{p-1})(6x6)$, $\Phi(t_o, t_N)(6x6)$

1. $\Phi(t_p, t_{p-1}) = A_6^{-1}(t_p)\, \Phi(t_p, t_{p-1})^S A_6(t_{p-1})$

2. When $t = t_N$ compute

   $\Phi(t_N, t_o) = A_6^{-1}(t_N)\, \Phi(t_N, t_o)\, A_6(t_o)$

3. $\Phi(t_o, t_N) = \Phi^{-1}(t_N, t_o)$

Block II. 5. 3 Transform Difference Equation Coefficients

Input: $A_6^{-1}(t_p)$ (6x6), $B^s_{p,p-1}$ (6x2), $C^s_{p,p-1}$ (6x1), $\Gamma^s_{p,p-1}$ (6x2)

Output: $B_{k,k-1}$ (6x2), $C_{k,k-1}$ (6x1), $\Gamma_{k,k-1}$ (6x2)

1. $B_{p,p-1} = A_6^{-1}(t_p)\, B^s_{p,p-1}$
2. $C_{p,p-1} = A_6^{-1}(t_p)\, C^s_{p,p-1}$
3. $\Gamma_{p,p-1} = A_6^{-1}(t_p)\, \Gamma^s_{p,p-1}$

Block II. 5. 4 Transform Acceleration Related Measurement Coefficients

Input: ${}_aJ_p^s$ (6x6), ${}_2J_p^s$ (6x2), ${}_3J_p^s$ (6x1), γ_p^s (6x2), $A_6^{-1}(t_k)$ (6x6)

Output: ${}_aJ_p$ (6x6), ${}_2J_p$ (6x2), ${}_3J_p$ (6x1), γ_p (6x2)

1. ${}_aJ_p = A_6^{-1}(t_p)\, {}_aJ_p^s \quad A_6(t_p)$

2. ${}_2J_p = A_6^{-1}(t_p)\, {}_2J_p^s$

3. ${}_3J_p = A_6^{-1}(t_p)\, {}_3J_p^s$

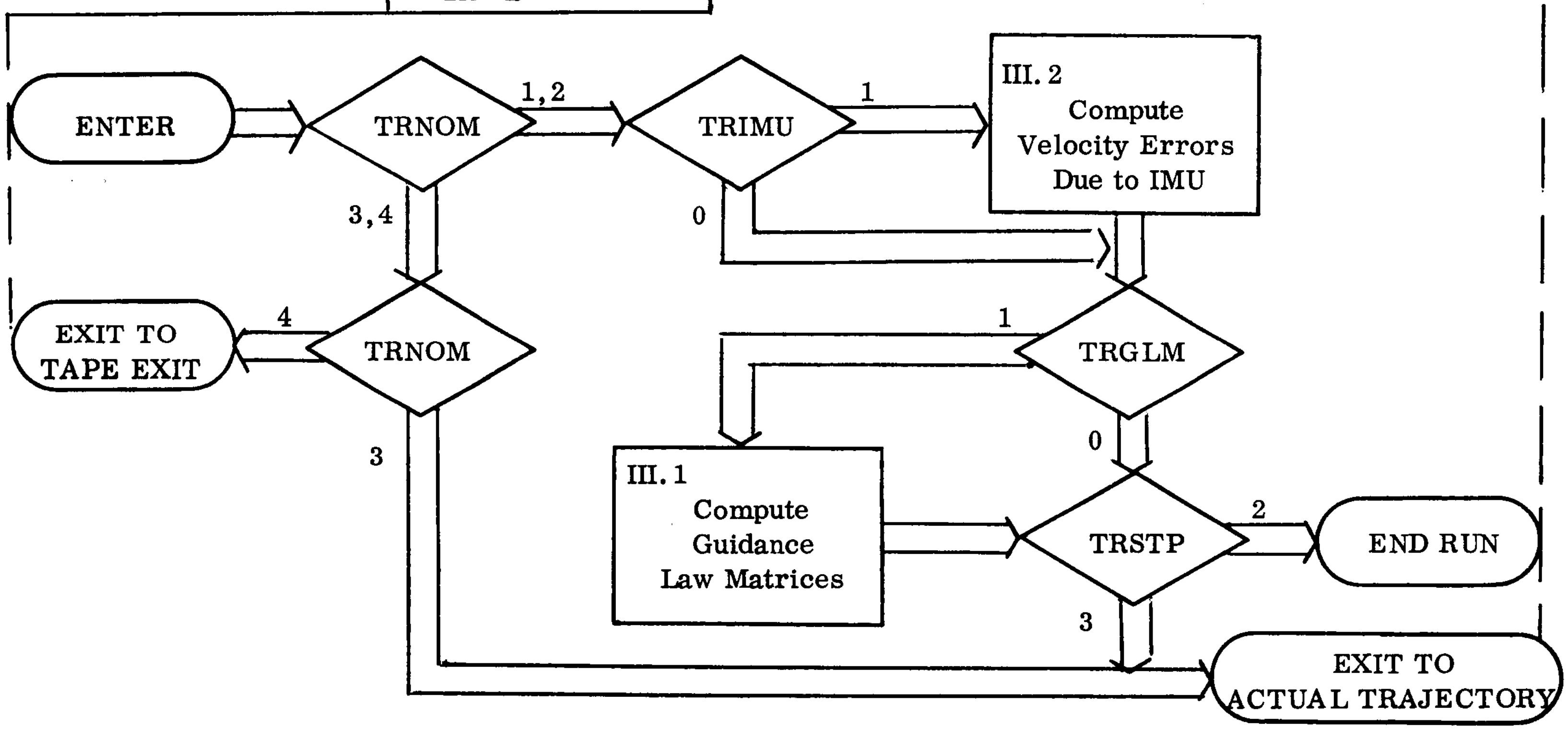
4. $\gamma_p = A_6^{-1}(t_p)\, \gamma_p^s$

Note: The upper three rows of all four matrices above are zeros and need not be computed and should not be stored.

3.4.3 Guidance Law and IMU Error Matrices

Stored	Computed
T_{ci}, Δt_{ci} $i = 1, 2, \ldots, 10$	$\Phi(t_{p-i}, t_{p-i-1})^{(6x6)}$
W_c^U tables, W'^X_c tables	$c^{\Phi}(t_{p-i}, t_{p-i-1})^{(1x1)}$
$M(3x3)$, K'_1 $(i = 1, 2, \ldots, 5)$	$B_{p-i, p-i-1}^{(6x2)}$
TRNOM, TRGLM,	$C_{p-i, p-i-1}^{(6x1)}$
TRIMU, TRSTP,	$\Gamma_{p-i, p-i-1}^{(6x2)}$
TROMG,	$c^{\Gamma}{}_{p-i, p-i-1}^{(1x1)}$
δt_{IM}, ϵ_I	a_x, a_y, a_z, ω_{PI}, ω_{RO},
	ω_{YA}, t_n

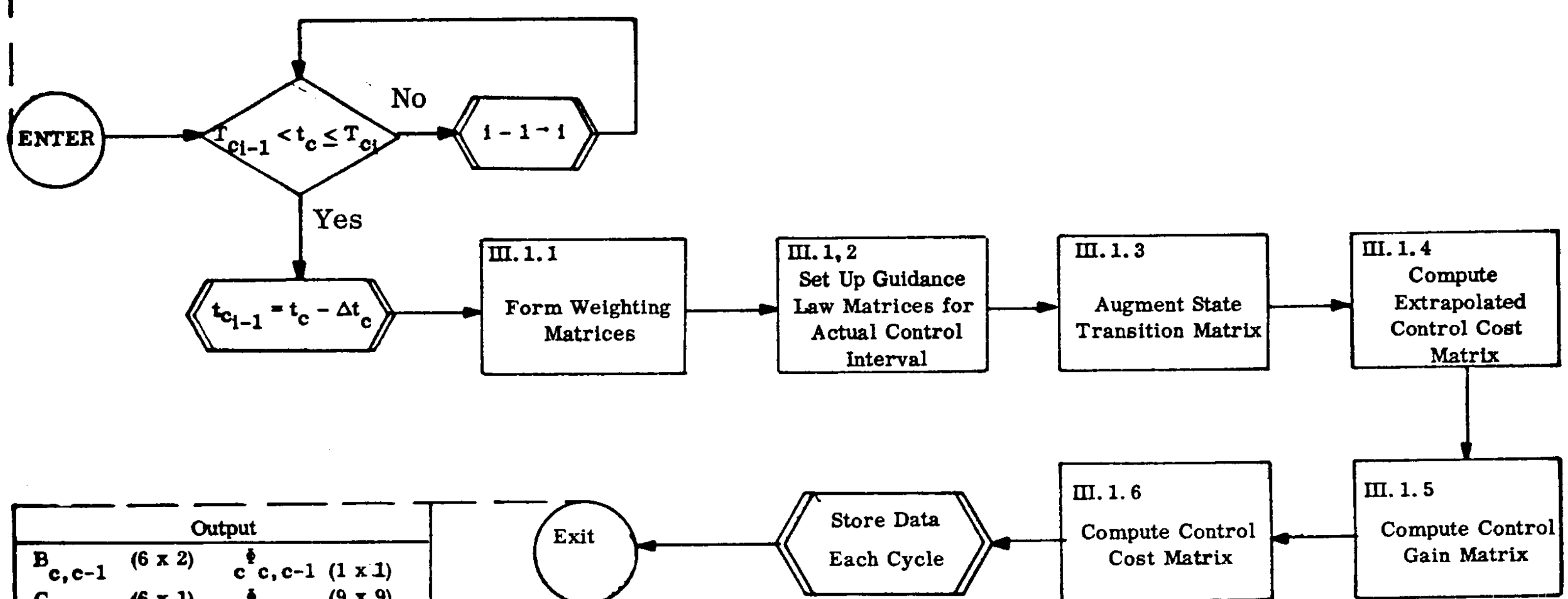
Output
$B_{c,c-1}^{(6x2)}$, $a^{\Phi}{}_{c,c-1}^{(9x9)}$
$C_{c,c-1}^{(6x1)}$, $\pi'_c{}^{(9x9)}$
$\Gamma_{c,c-1}^{(6x2)}$, $\Lambda_c^{(2x9)}$
$c^{\Gamma}{}_{c,c-1}^{(1x1)}$, $\pi_{c t_p}^{(9x9)}$
$c^{\Phi}{}_{c,c-1}^{(1x1)}$, $\int_{t_o} G dt$



3.4.3.1 Level II Flow Chart - Guidance Law and IMU Error Matrices

3.4.3.2 Detailed Flow Charts and Equations

Stored	Computed
T_{c_i}, Δt_{c_i} $i = 1,2,\ldots,10$	$\Phi(t_{p-i}, t_{p-i-1})$ (6 x 6)
W_c^U tables	$c^{\Phi}(t_{p-i}, t_{p-i-1})$ (1 x 1)
W_c^X tables	$\Gamma_{p-i,p-i-1}$ (6 x 2)
	$c^{\Gamma}_{p-i,p-i-1}$ (1 x 1)



Output			
$B_{c,c-1}$	(6 x 2)	$c^{\Phi}_{c,c-1}$	(1 x 1)
$C_{c,c-1}$	(6 x 1)	$a^{\Phi}_{c,c-1}$	(9 x 9)
$\Gamma_{c,c-1}$	(6 x 2)	π'	(9 x 9)
$c^{\Gamma}_{c,c-1}$	(1 x 1)	Λ	(2 x 9)
		π	(9 x 9)

3.4.3.2.1 Compute Guidance Law Matrices - Block III.1

Block III. 1. 1 Form Weighting Matrices

Input: Table W^U, Table W^X

Output: W^U_{c-1} (2x2), W^X_c (9x9)

The matrices W^U_{c-1} and W^X_c are both symmetric matrices. The elements of these matrices are tabulated functions of fifty time points. The values of the elements in the matrices are used in the time interval $t_{j-1} \leq t < t_j$, $j = 1, 2, \ldots, 10$.

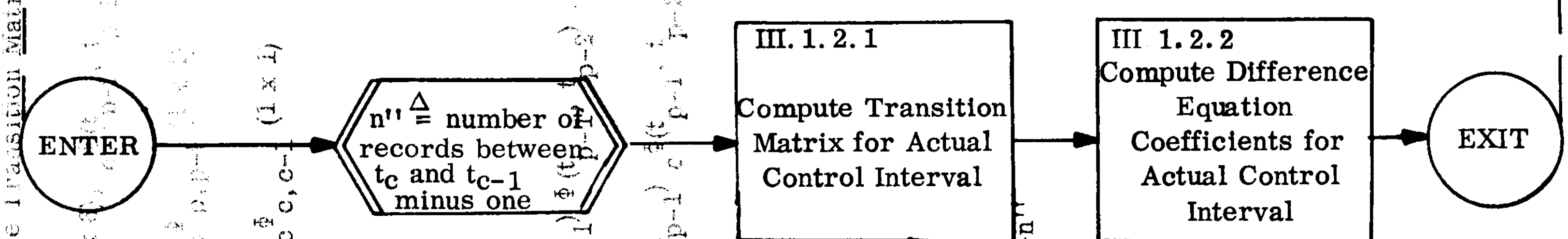
1) Form W^U_{c-1}

2) Form W^X_c using the elements in the table to generate W'^X_c (6 x 6) matrix and fill the remainder of the 9 x 9 matrix with zeros.

$$\text{The form of } W^X_c = \begin{bmatrix} W'^X_c & 0 \\ 0 & 0 \end{bmatrix}$$

Stored	Computed
T_{ci}, ΔT_{ci} $i = 1,2,\ldots,10$	$\Phi(t_{p-i}, t_{p-i-1})$ (6 x 6) $c^{\Phi}(t_{p-i}, t_{p-i-1})$ (1 x 1) $B_{p-i,p-i-1}$ (6 x 2) $C_{p-i,p-i-1}$ (6 x 1) $\Gamma_{p-i,p-i-1}$ (6 x 2) $c^{\Gamma}{}_{p-i,p-i-1}$ (1 x 1)

Output	
$\Phi_{c,c-1}$	(6 x 6)
$c^{\Phi}{}_{c,c-1}$	(1 x 1)
$B_{c,c-1}$	(6 x 2)
$C_{c,c-1}$	(6 x 1)
$\Gamma_{c,c-1}$	(6 x 2)
$c^{\Gamma}{}_{c,c-1}$	(1 x 1)



Block III. 1. 2 Set Up Guidance Law Matrices for Actual Control Interval

AC ELECTRONICS DIVISION GENERAL MOTORS CORPORATION

Block III. 1. 2. 2 Compute Difference Equation Coefficients for Actual Control Interval

Input: $\Phi_{p,p-i}$ (6 x 6), ${}_c\Phi_{p,p-i}$ (1 x 1), $B_{p-i,\ p-i-1}$ (6 x 2), $C_{p-i,p-i-1}$ (6 x 1),

Output: $B_{c,c-1}$ (6 x 2), $C_{c,c-1}$ (6 x 1), $\Gamma_{c,c-1}$ (6 x 2), ${}_c\Gamma_{c,c-1}$ (1 x 1)

1) $$B_{c,c-1} = \sum_{i=0}^{n''-1} \Phi_{p,p-i}\, B_{p-i,\ p-i-1}$$

2) $$C_{c,c-1} = \sum_{i=0}^{n''-1} \Phi_{p,p-i}\, C_{p-i,p-i-1}$$

3) $$\Gamma_{c,c-1} = \sum_{i=0}^{n''-1} \Phi_{p,p-i}\, \Gamma_{p-i,p-i-1}$$

4) $${}_c\Gamma_{c,c-1} = \sum_{i=0}^{n''-1} [{}_c\Phi_{p,p-i}]\,[{}_c\Gamma_{p-i,p-i-1}]$$

where $i = 0,1,2,\ldots,n''-1$

Block III. 1. 3 Augment State Transition Matrix

Input: $\Phi_{c+1,c}$ (6 x 6) $B_{c+1,c}$ (6 x 2) $C_{c+1,c}$ (6 x 1) ${}_c\Phi_{c+1,c}$ (1 x 1)

0_1 (2 x 6), 0_2 (2 x 1), 0_3 (1 x 6), 0_4 (1 x 2), I (2 x 2)

Output: ${}_a\Phi_{c+1,c}$ (9 x 9)

$$
{}_a\Phi_{c+1,c} = \begin{bmatrix} \Phi_{c+1,c} & B_{c+1,c} & C_{c+1,c} \\ 0_1 & I & 0_2 \\ 0_3 & 0_4 & {}_c\Phi_{c+1,c} \end{bmatrix}
$$

Block III. 1. 4 Computed Extrapolated Control Cost Matrix

Input: ${}_a\Phi_{c+1,c}$ (9x9), π_{c+1} (9x9), W_c^X (9x9)

Output: π'_c (9x9)

The following matrix multiplication is performed

1) $$\pi'_c = [{}_a\Phi^T_{c+1,c}][\pi_{c+1}][{}_a\Phi_{c+1,c}] + [W_c^X]$$

Equation 1 is rewritten below to indicate the fact that ${}_a\Phi_{c+1,c}$ has four zero submatrices which may be pertinent to the method used to code the equation.

The π matrix may be partitioned as follows.

2) $$\pi_{c+1} \triangleq \begin{bmatrix} {}_1\pi & {}_2\pi & {}_3\pi \\ {}_4\pi & {}_5\pi & {}_6\pi \\ {}_7\pi & {}_8\pi & {}_9\pi \end{bmatrix}$$

3) $$\pi'_c = \begin{bmatrix} \Phi^T & 0_1^T & 0_3^T \\ B^T & I & 0_4^T \\ C^T & 0_2^T & {}_c\Phi \end{bmatrix} \begin{bmatrix} {}_1\pi & {}_2\pi & {}_3\pi \\ {}_4\pi & {}_5\pi & {}_6\pi \\ {}_7\pi & {}_8\pi & {}_9\pi \end{bmatrix} \begin{bmatrix} \Phi & B & C \\ 0_1 & I & 0_2 \\ 0_3 & 0_4 & {}_c\Phi \end{bmatrix} + \left[W_c^X \right]$$

The dimensions of the submatrices are defined below.

Dim $[0_1]$	=	(2 x 6)	Dim $[{}_1\pi]$	=	(6 x 6)	Dim $[{}_6\pi]$	=	(2 x 1)
Dim $[0_2]$	=	(2 x 1)	Dim $[{}_2\pi]$	=	(6 x 2)	Dim $[{}_7\pi]$	=	(1 x 6)
Dim $[0_3]$	=	(1 x 6)	Dim $[{}_3\pi]$	=	(6 x 1)	Dim $[{}_8\pi]$	=	(1 x 2)
Dim $[0_4]$	=	(1 x 2)	Dim $[{}_4\pi]$	=	(2 x 6)	Dim $[{}_9\pi]$	=	(1 x 1)
Dim $[I]$	=	(2 x 2)	Dim $[{}_5\pi]$	=	(2 x 2)			

Block III. 1. 5 Compute Control Gain Matrix

Input: $\Gamma_{c,c-1}$ (6 x 2), π'_c (9 x 9), W^U_{c-1} (2 x 2)

Output: Λ_c (2 x 9)

1) $${}_a\Gamma_{c,c-1} = \begin{bmatrix} \Gamma_{c,c-1} \\ \\ {}_1 0 \end{bmatrix} \quad ; \text{ Dim } [{}_1 0] = (3 \times 2)$$

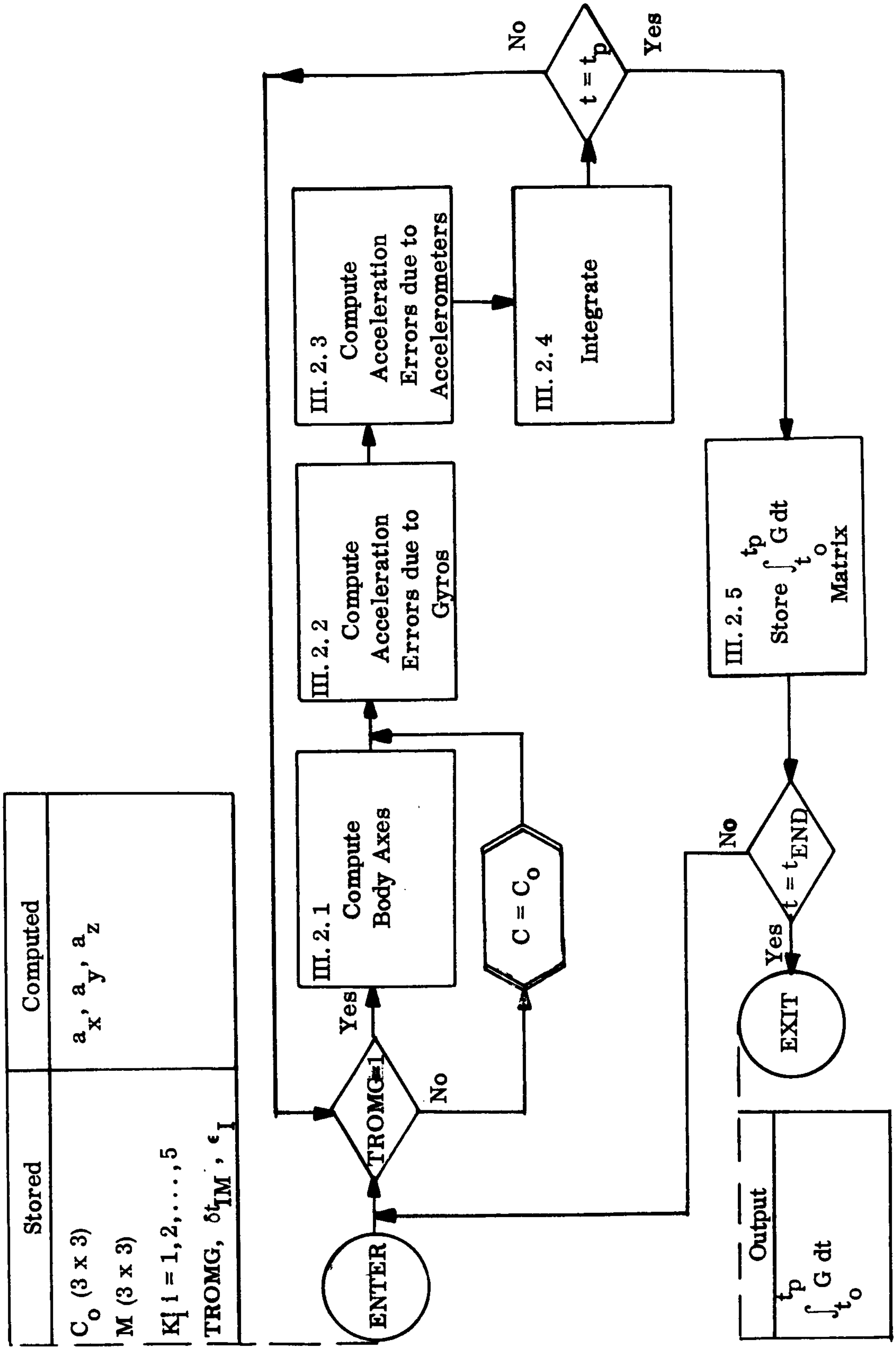
2) $$\Lambda_c = \{ [{}_a\Gamma^T_{c,c-1}][\pi'_c] \; [{}_a\Gamma_{c,c-1}] + W^U_{c-1} \}^{-1} [{}_a\Gamma^T_{c,c-1}][\pi'_c]$$

Block III. 1. 6 Compute Control Gain Matrix

Input: π'_c (9 x 9), ${}_a\Gamma_{c,\,c-1}$(9 x 2), Λ_c (2 x 9)

Output: π_c (9 x 9)

1) $\pi_c = [\pi'_c] - [\pi'_c]\,[{}_a\Gamma_{c,\,c-1}][\Lambda_c]$



3.4.3.2.2 Compute Velocity Errors Due to IMU - Block III.2

Block III. 2. 1 Compute Body Axes

Input: C_o (3x3), $\alpha_1, \alpha_2. \alpha_3$

Output: C (3x3)

1. $$[C'] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_3 & \sin\alpha_3 \\ 0 & -\sin\alpha_3 & \cos\alpha_3 \end{bmatrix} \begin{bmatrix} \cos\alpha_2 & 1 & -\sin\alpha_2 \\ 0 & 1 & 0 \\ \sin\alpha_2 & 0 & \cos\alpha_2 \end{bmatrix} \begin{bmatrix} \cos\alpha_1 & \sin\alpha_1 & 0 \\ -\sin\alpha_1 & \cos\alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2. $$[C] = [C][C_o]$$

Block III. 2. 2 Compute Acceleration Errors due to Gyros

Input: M (3 x 3), a_x, a_y, a_z, C (3 x 3), t, $\int_{t_o}^{t} a_1$, $\int_{t_o}^{t} a_2$, $\int_{t_o}^{t} a_3$, K_1', K_2', K_3'

Output: G_{i1} (3 x 3), a_i $\qquad$ i = 1, 2, 3

1) $$\underline{f} \triangleq \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$$

2) $$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = M\, C\, \underline{f}$$

3) $$M\ (3 \times 3) \triangleq \begin{bmatrix} m_1\ (1 \times 3) \\ m_2\ (1 \times 3) \\ m_3\ (1 \times 3) \end{bmatrix}$$

4) $$M_1 \triangleq \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \ ; \ M_2 \triangleq \begin{bmatrix} m_2 \\ m_3 \\ m_1 \end{bmatrix} \ ; \ M_3 \triangleq \begin{bmatrix} m_3 \\ m_1 \\ m_2 \end{bmatrix}$$

5) $$G_{i1} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \left\{ C'^{T}_{o}\, M_i^T \begin{bmatrix} K_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \int_{t_o}^{t} C^T M_i^T \begin{bmatrix} 0 & K_2 & K_3 a_i \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} dt \right\}$$

Block III. 2. 3 Compute Acceleration Errors due to Accelerometers

Input: C^T (3 x 3), M_i (3 x 3), a_i, K'_4, K'_5 i = 1, 2, 3

Output: G_{i2} (3 x 2) i = 1, 2, 3

1) $$G_{i2} = C^T M_i \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} [\, 1 \; a_i \,] \begin{bmatrix} K'_4 & 0 \\ 0 & K'_5 \end{bmatrix}$$

i = 1, 2, 3

Block III. 2. 4 Integrate

Input: $\dot{C}$ (3 x 3), G_{i1} (3 x 3), G_{i2} (3 x 2), a_i, i = (1, 2, 3), δt_{IM}, ϵ_I

Output: $\int_{t_o}^{t} G_{i1} d\tau$ (3 x 3), $\int_{t_o}^{t} G_{i2} d\tau$ (3 x 2) $\int_{t_o}^{t} a_i d\tau$ (3 x 1) (i = 1, 2, 3)

The integration routine used in this section is the same Runge-Kutta routine which is used in the nominal and actual trajectory blocks. The integration step size is the minimum of the input value, δt_{IM}, and the interval between nominal control times. A linear interpolation of data at t_G time points is made to obtain data between these time points when required as input to the integration routine.

Block III.2.5 Store $\int_{t_o}^{t_p} G\, dt$ Matrix

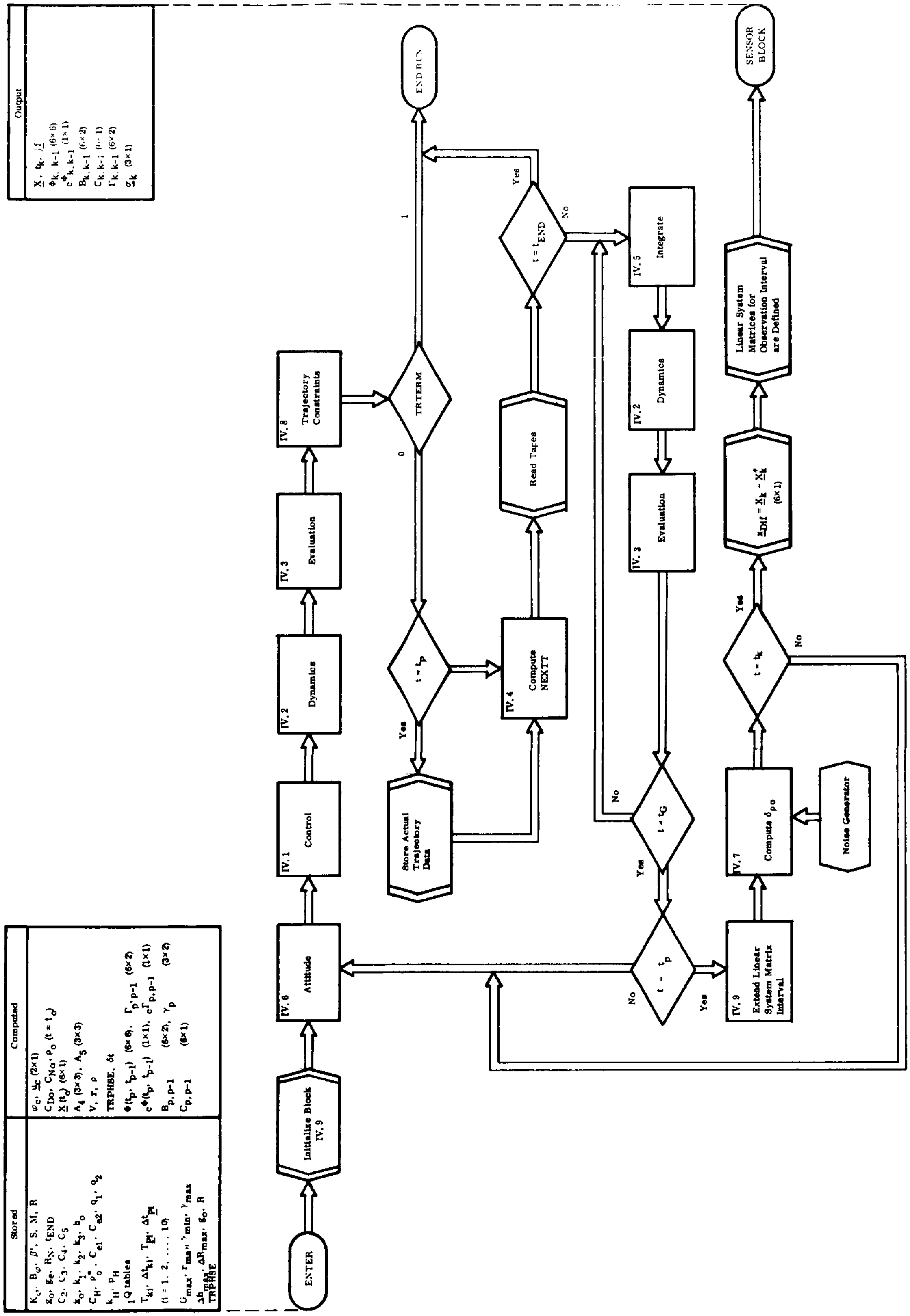
Input: $\int_{t_o}^{t_p} G_{i1}\, d\tau$ (3 x 3), $\int_{t_o}^{t_p} G_{i2}\, d\tau$ (3 x 2) $\qquad i = 1, 2, 3$

Output: $\int_{t_o}^{t_p} G\, dt$ (3 x 15)

The nominal observation times, t_p, are defined by the tape generated in Blocks I and II containing the input to Block III. 2. At each of these times $\int_{t_o}^{t_p} G\, dt$ is formed as indicated below and stored on tape for use if needed in the sensor block.

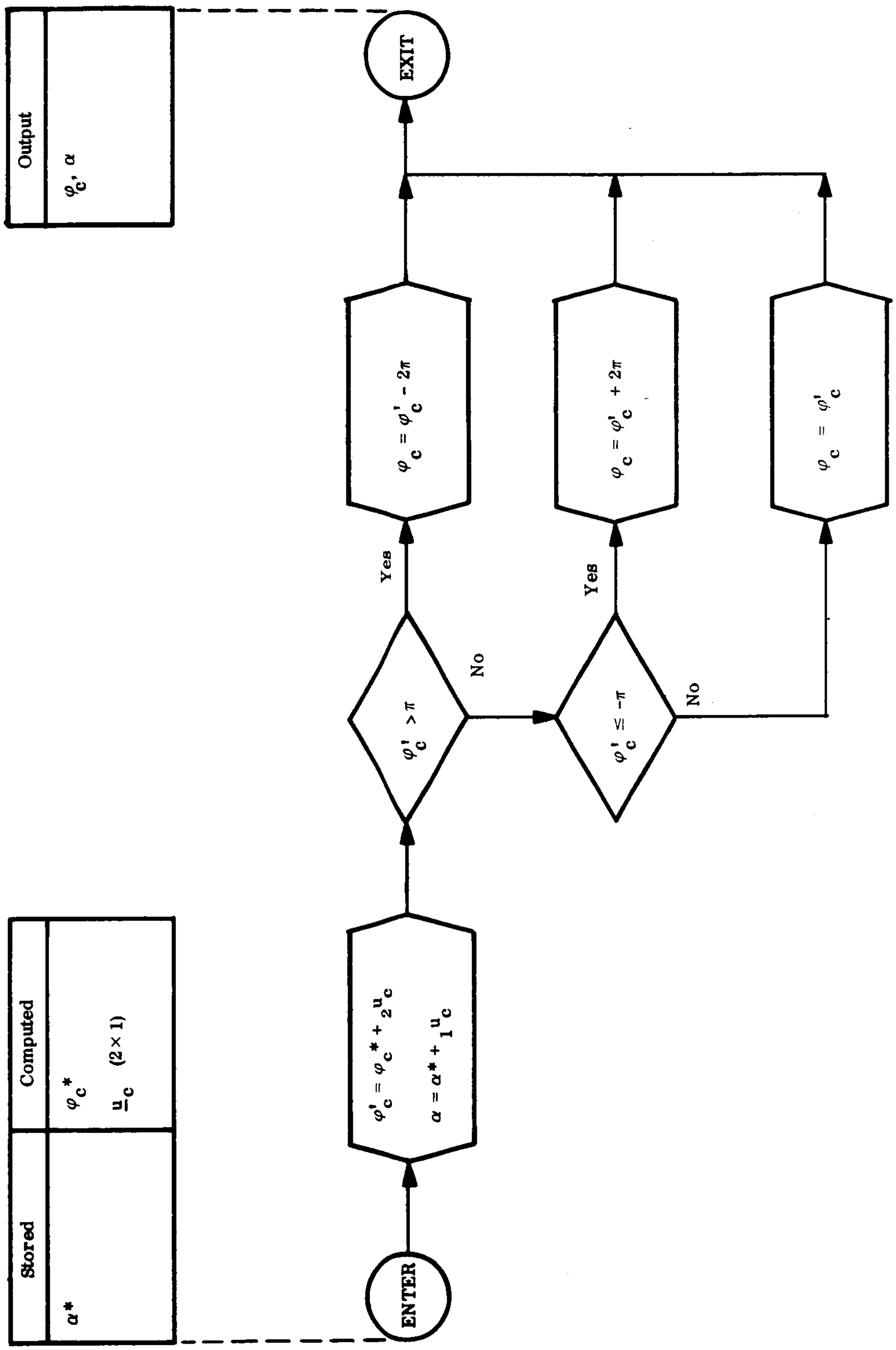
1) $$\int_{t_o}^{t_p} G\, dt = \left[\int_{t_o}^{t_p} G_{11}\, dt \quad \int_{t_o}^{t_p} G_{21}\, dt \quad \int_{t_o}^{t_p} G_{31}\, dt \quad \int_{t_o}^{t_p} G_{12}\, dt \quad \int_{t_o}^{t_p} G_{22}\, dt \quad \int_{t_o}^{t_p} G_{32}\, dt \right]$$

3.4.4 Actual Trajectory

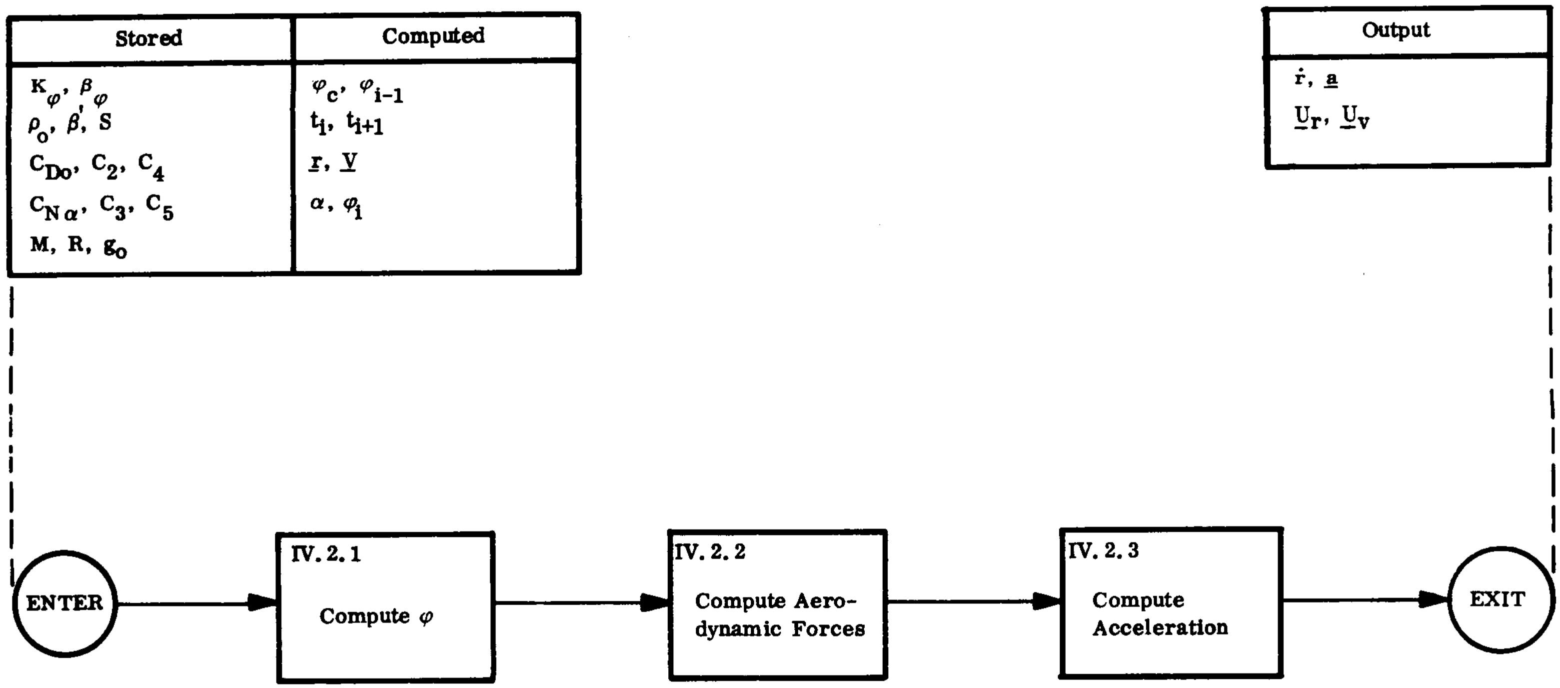


3.4.4.1 Level II Flow Chart - ACTUAL Trajectory

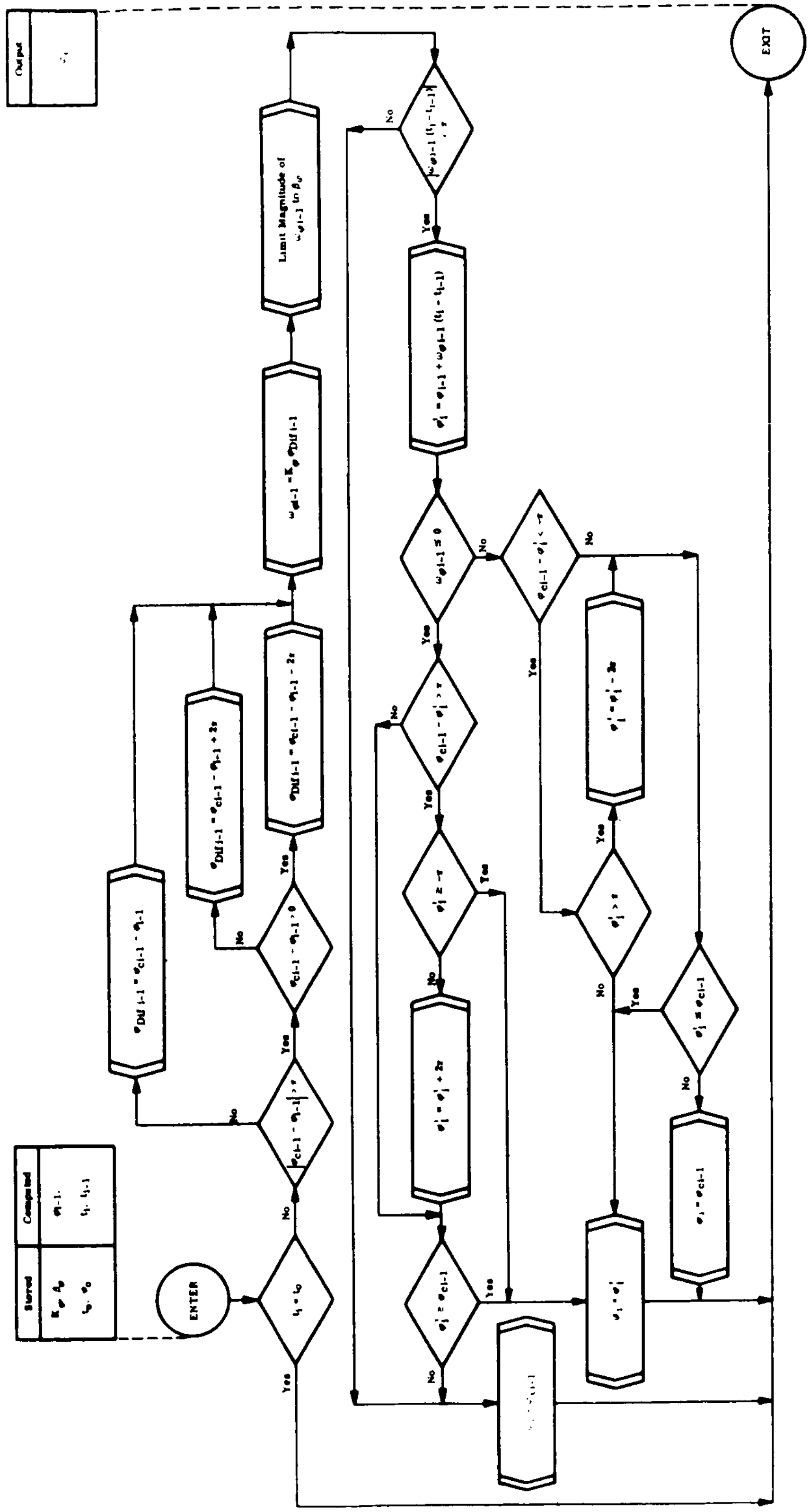
3.4.4.2 Detailed Flow Charts and Equations



3.4.4.2.1 Control - Block IV.1



3.4.4.2.2 Dynamics - Block IV.2



Block IV.2.1 - Compute φ

Block IV.2.2 Compute Aerodynamic Forces

INPUT: $\underline{r}$, $\underline{V}$, ρ_o, β', R, S, C_{Do}, C_2, C_4, $C_{N\alpha}$, C_3, C_5, α, φ_i

OUTPUT: $\underline{U}_v$, $\underline{U}_r$, $\underline{U}_{po}$, $\underline{D}$, $\underline{N}$, $\dot{r}$

1. $V = +\sqrt{\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2}$

2. $\underline{U}_v = \dfrac{\underline{V}}{V}$

3. $r = +\sqrt{X^2 + Y^2 + Z^2}$

4. $\underline{U}_r = \dfrac{\underline{r}}{r}$

5. $\gamma = \sin^{-1} [\underline{U}_r \cdot \underline{U}_v]$

6. $\dot{r} = V \sin \gamma$

7. $\underline{U}_u = \dfrac{\underline{U}_r - \underline{U}_v \sin \gamma}{\cos \gamma}$

8. $\underline{U}_p = \underline{U}_u \times \underline{U}_v$

9. $\rho = (\rho_o + \delta\rho_o)\, e^{-\beta'(r - R)}$

10. $C_D = C_{Do} + C_2\alpha^2 + C_4\alpha^4$

11. $C_N = C_{N\alpha} + C_3\alpha^3 + C_5\alpha^5$

12. $\underline{D} = - C_D \rho \dfrac{V^2 S}{2} \underline{U}_v$

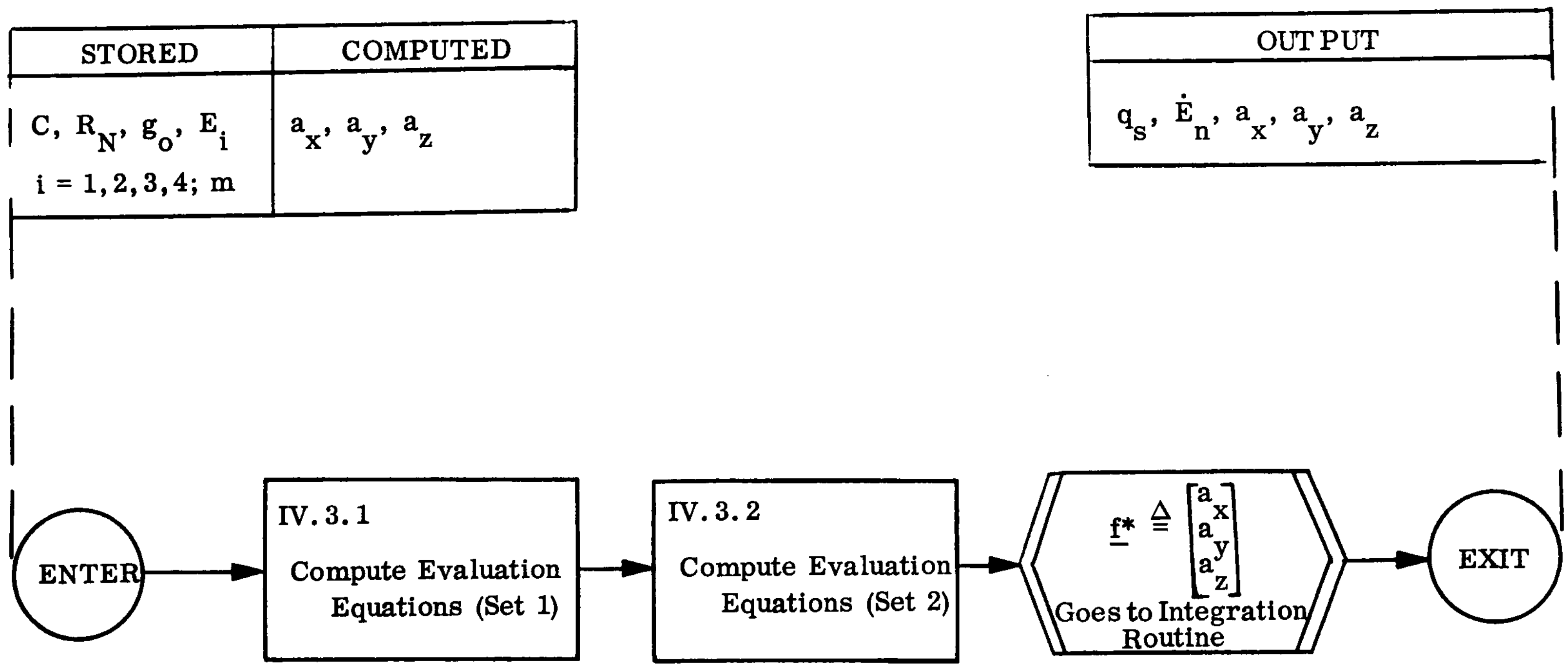
13. $\underline{N} = C_N \rho \dfrac{V^2 S}{2} [\cos \varphi_i \underline{U}_u - \sin \varphi_i \underline{U}_p]$

Block IV.2.3 Compute Acceleration

INPUT: $\underline{D}$, $\underline{N}$, M, R, g_0

OUTPUT: a_x, a_y, a_z, $\ddot{X}$, $\ddot{Y}$, $\ddot{Z}$, $\underline{a}$, $\underline{f}$

1. $a_x = (D_x + N_x)/M$
2. $a_y = (D_y + N_y)/M$
3. $a_z = (D_z + N_z)/M$
4. $g = g_0 \left(\frac{R}{r}\right)^2$
5. $\ddot{X} = a_x - g \frac{X}{r}$
6. $\ddot{Y} = a_y - g \frac{Y}{r}$
7. $\ddot{Z} = a_z - g \frac{Z}{r}$
8. $\underline{a} \triangleq \ddot{X}\, \underline{i} + \ddot{Y}\, \underline{j} + \ddot{Z}\, \underline{k}$
9. $\underline{f} \triangleq a_x \underline{i} + a_y \underline{j} + a_z \underline{k}$



3.4.4.2.3 Evaluation - Block IV.3

Block IV.3.1 Compute Evaluation Equations (Set 1)

INPUT: C_H, R_N, ρ, ρ_o, g, g_e, C_{e1}, C_{e2}, q_1, q_2, k_H, p_H, V, r, E_i

OUTPUT: q_s, E_n, a'

1. $$q_c = \frac{C_H}{\sqrt{R_N}} \left(\frac{\rho}{\rho_o}\right)^n \left(\frac{V}{\sqrt{g\,r}}\right)^m$$

2. If $\frac{V}{\sqrt{g\,r}} < 1.73$: $q_1 \rightarrow q$; $C_{e1} \rightarrow C_e$

   If $\frac{V}{\sqrt{g\,r}} \geq 1.73$: $q_2 \rightarrow q$; $C_{e2} \rightarrow C_e$

3. $$q_r = k_H R_N \left(\frac{\rho}{\rho_o}\right)^{p_H} C_e V^q$$

4. $$q_s = q_c + q_r$$

5. $$a' = \frac{\sqrt{a_x^2 + a_y^2 + a_z^2}}{g_e}$$

6. $$\tau' = E_o + E_1(a') + E_2(a')^2 + E_3(a')^3 + E_4(a')^4$$

7. $$\dot{E}'_n = \frac{1}{\tau'}$$

8. Is $\dot{E}'_n \leq 0.0008$?

   a. Yes: $\dot{E}_n = 0$

   b. No: $\dot{E}_n = \dot{E}'_n$

Block IV. 3. 2 Compute Evaluation Equations (Set 2)

INPUT: $X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}, f, r, g_e, R, A_5$

OUTPUT: $\theta, \phi, \beta, a', h$

1. $\underline{r}_t = A_5 \underline{r}$

2. $\underline{V}_t = A_5 \underline{V}$

3. $\theta = \cos^{-1}\left[\frac{Z_t}{r}\right]$

   a. If $Y_t \geq 0$ then $0 \leq \theta \leq \pi$

   b. If $Y_t < 0$ then $\pi < \theta < 2\pi$

4. $\phi' = \tan^{-1}\left[\frac{X_t}{Y_t}\right]$ $\quad -\pi < \phi \leq \pi$

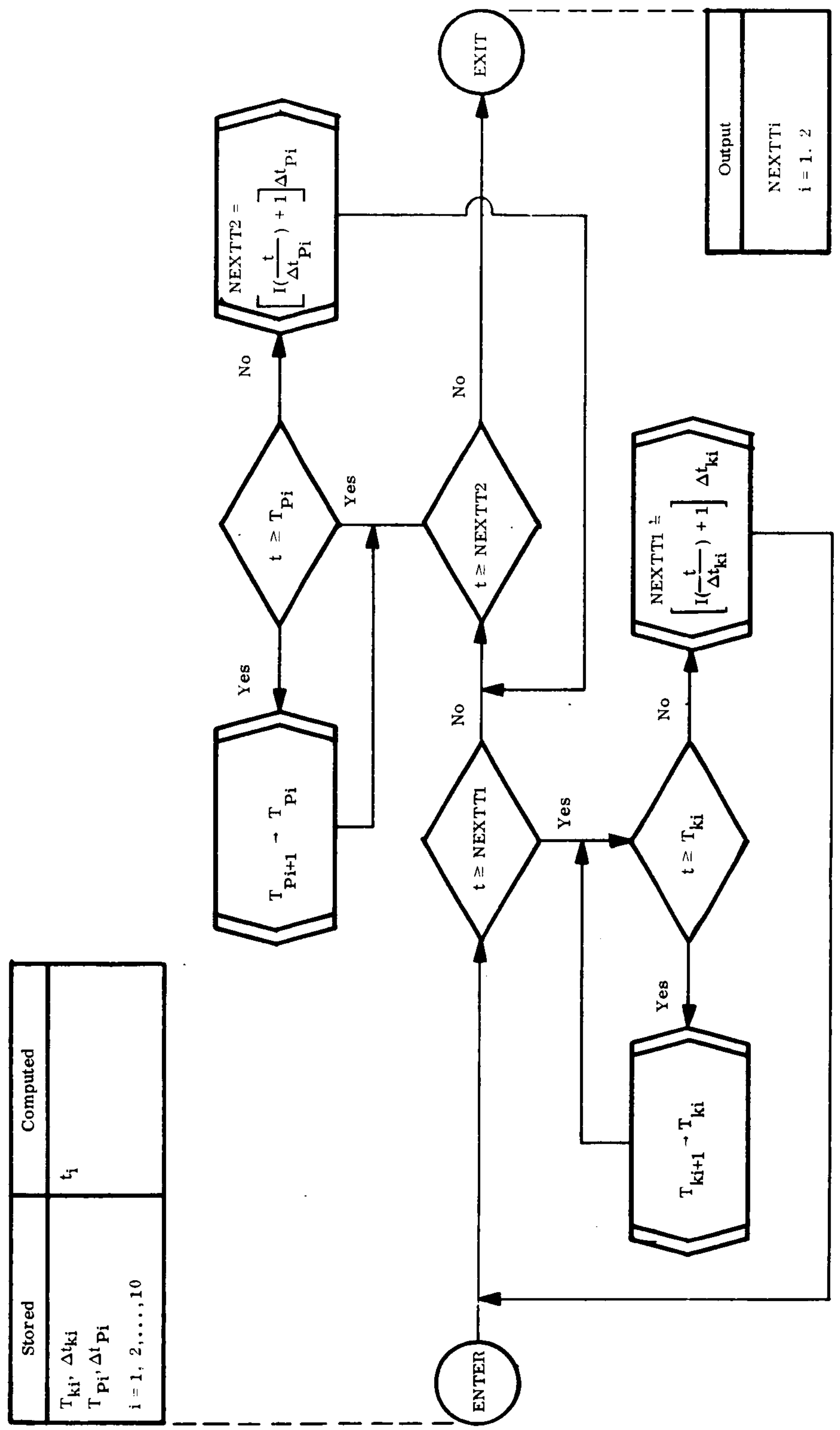
   a. If $\frac{\sqrt{X_t^2 + Y_t^2}}{r} < 0.015$ then $\phi_i = \phi_{i-1}$

   b. If $\frac{\sqrt{X_t^2 + Y_t^2}}{r} \geq 0.015$ then $\phi_i = \phi'_i$

5. $\beta = \tan^{-1}\left[\frac{\cos\phi\,\dot{X}_t - \sin\phi\,\dot{Y}_t}{\cos\theta\sin\phi\,\dot{X}_t + \cos\theta\cos\phi\,\dot{Y}_t - \sin\theta\,\dot{Z}_t}\right]$ $\quad -\pi < \beta < \pi$

6. $a' = \frac{f}{g_e}$

7. $h = (r - R)$



3.4.4.2.4 Compute NEXTT - Block IV.4

3.4.4.2.5 Integrate - Block IV.5

Input: $\ddot{X}$, $\ddot{Y}$, $\ddot{Z}$, $\dot{X}$, $\dot{Y}$, $\dot{Z}$, q_s, $\dot{E}_n$, $\underline{f}_t$

Output: X, Y, Z, $\dot{X}$, $\dot{Y}$, $\dot{Z}$, Q_H, E_n, $\int_{t_o} \underline{f}\, dt$

The integration routine uses the same fixed step size as that used in the nominal trajectory block and the integration routine is the same.

3.4.4.2.6 Attitude - Block IV.6

INPUT: $\varphi, \alpha, \underline{U}_v, \underline{U}_p, \underline{U}_u, \underline{P}_{Io}, \underline{Y}_{Ao}, \underline{R}_{Oo}, t_i, t_{i-1}$

OUTPUT: $\alpha_1, \alpha_2, \alpha_3, \omega_{PI}, \omega_{YA}, \omega_{RO}$

$$\begin{bmatrix} \underline{P}_I \\ \underline{Y}_A \\ \underline{R}_O \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \underline{U}_v \\ \underline{U}_p \\ \underline{U}_u \end{bmatrix}$$

$$\alpha_1 = \tan^{-1}\left[\frac{\underline{P}_I \cdot \underline{Y}_{Ao}}{\underline{P}_I \cdot \underline{P}_{Io}}\right] \qquad -\pi < \alpha_1 \leq \pi$$

$$\alpha_2 = \sin^{-1}\left[\underline{P}_I \cdot \underline{R}_{Oo}\right] \qquad -\frac{\pi}{2} \leq \alpha_2 \leq \frac{\pi}{2}$$

$$\alpha_3 = \tan^{-1}\left[\frac{\underline{Y}_A \cdot \underline{R}_{Oo}}{\underline{R}_O \cdot \underline{R}_{Oo}}\right] \qquad -\pi < \alpha_3 \leq \pi$$

$$\left.\begin{aligned} \dot{\alpha}_1 &= \frac{\alpha_{1i} - \alpha_{1(i-1)}}{t_i - t_{i-1}} \\ \dot{\alpha}_2 &= \frac{\alpha_{2i} - \alpha_{2(i-1)}}{t_i - t_{i-1}} \\ \dot{\alpha}_3 &= \frac{\alpha_{3i} - \alpha_{3(i-1)}}{t_i - t_{i-1}} \end{aligned}\right\} \qquad \dot{\alpha}_1 = \dot{\alpha}_2 = \dot{\alpha}_3 = 0 \qquad \text{at } t = t_o$$

$$\omega_{RO} = \cos\alpha_2 \cos\alpha_3(\dot{\alpha}_1) - \sin\alpha_3(\dot{\alpha}_2)$$

$$\omega_{YA} = \cos\alpha_2 \sin\alpha_3(\dot{\alpha}_1) + \cos\alpha_3(\dot{\alpha}_2)$$

$$\omega_{PI} = -\sin\alpha_2(\dot{\alpha}_1) + \dot{\alpha}_3$$

3.4.4.2.7 Compute $\delta\rho_o$ - Block IV.7

Input: k_o, k_1, k_2, k_3, $\dot{h}(t_{p-1})$, h_o, $_c\Phi(t_p, t_{p-1})$, $\delta\rho_o(t_{p-1})$, $_c\Gamma_{p,p-1}$

Output: $\delta\rho_o(t_p)$

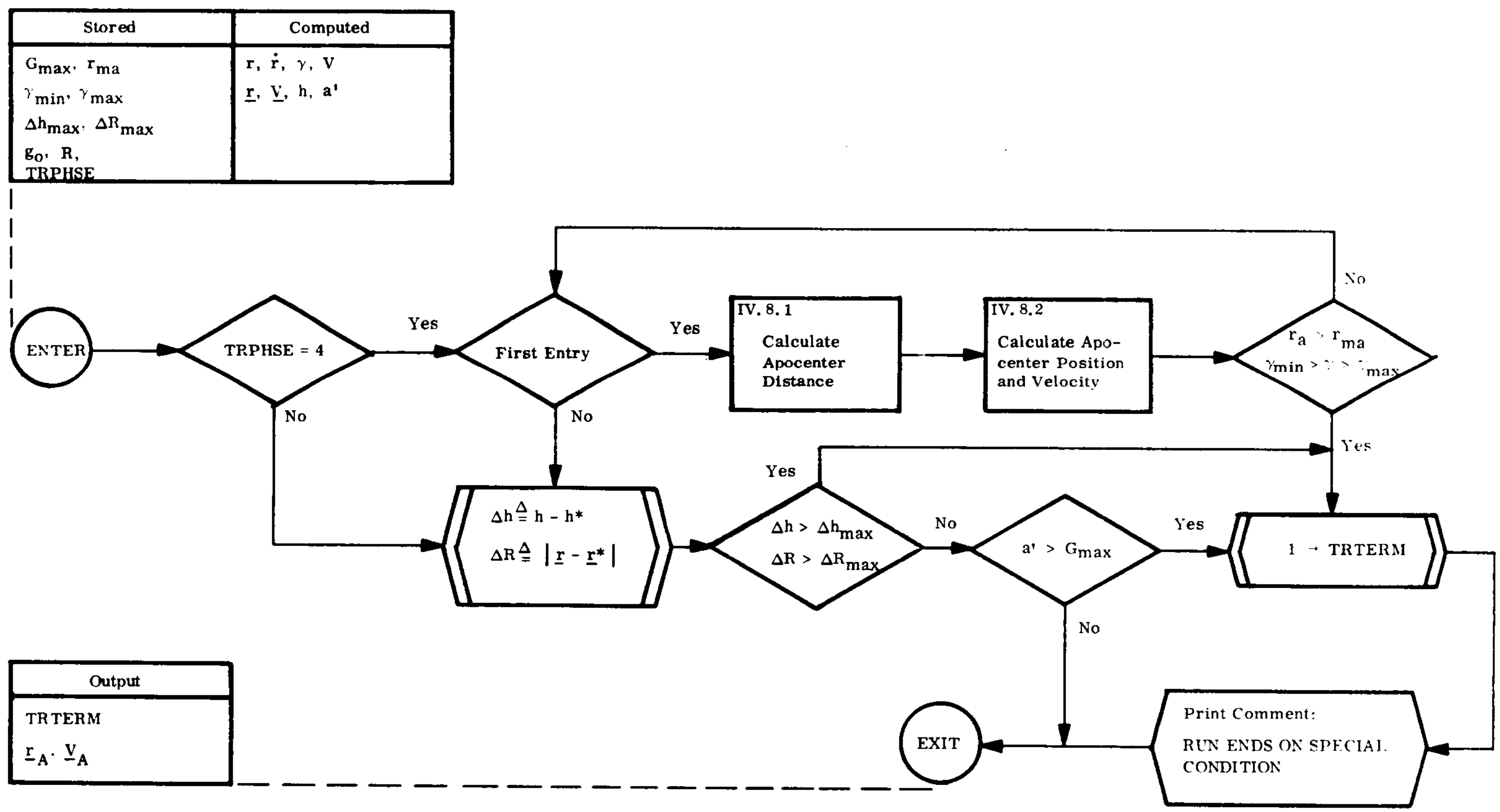
1. $${}_2Q_{p-1} = |\dot{h}(t_{p-1})| (k_o + [k_1 + k_2 h(t_{p-1})] e^{-k_3 [h(t_{p-1}) - h_o]}$$

2. Using the noise generator and $_2Q_{p-1}$ as the variance, generate a gaussian random number with zero mean $w_\rho(t_{p-1})$.

3. $$\delta\rho_o(t_p) = {}_c\Phi(t_p, t_{p-1})\, \delta\rho_o(t_{p-1}) + {}_c\Gamma_{p,p-1}\, w_\rho(t_{p-1})$$

This value of $\delta\rho_o$ is used with ρ_o until a new value is computed.

At $t = t_o$, $\delta\rho_o$ is computed in the initialization blocks.



3.4.4.2.8 Trajectory Constraint - Block IV.8

Block IV.8.1 Calculate Apocenter, Pericenter Distances

Input: g_o, R, r, V, γ

Output: r_a, r_p

1. $\mu = g_o R^2$

2. $p = \frac{(r V \cos \gamma)^2}{\mu}$

3. $a_e = \frac{r\mu}{2\mu - r V^2}$

4. $e = +\sqrt{1 - \frac{p}{a_e}}$ limit $\frac{p}{a_e}$ to ≤ 1

5. Is $a_e < 0$?

   a. Yes: $r_a = 10^{20}$

   b. No: $r_a = a(1 + e)$

6. $r_p = a_e(1 - e)$

Block IV. 8. 2 Calculate Apocenter Position and Velocity

Input: r, $\dot{r}$, $\underline{r}$, $\underline{V}$, V, μ, $\underline{U}_p$, a_e

Output: $\underline{r}_a$, $\underline{V}_a$

1. $$\underline{P} = \frac{1}{\mu e}\left[\left(V^2 - \frac{\mu}{r}\right)\underline{r} - (r\,\dot{r})\,\underline{V}\right]$$

2. $$\underline{r}_a = -\,r_a\,\underline{P}$$

3. $$V_a = +\sqrt{\mu\left(\frac{2}{r_a} - \frac{1}{a_e}\right)}$$

4. $$\underline{V}_a = V_a\,\underline{P} \times \underline{U}_p$$

3.4.4.2.9 Extend Linear System Matrix Interval - Block IV.9

Input: $\Phi_{p,p-1}(6x6)$, $c^{\Phi}{}_{p,p-1}(1x1)$, $B_{p,p-1}(6x2)$, $C_{p,p-1}(6x1)$, $\Gamma_{p,p-1}(6x2)$,

$c^{\Gamma}{}_{p,p-1}(1x1)$, $\underline{u}_{k-1}$, $\underline{\sigma}_{k-1}(3x1)$

Output: $\Phi_{p,k-1}(6x6)$, $c^{\Phi}{}_{p,k-1}(1x1)$, $B_{p,k-1}(6x2)$, $C_{p,k-1}(6x1)$, $\Gamma_{p,k-1}(6x2)$,

$\underline{\sigma}_p$ (3x1)

1. $B_{p,k-1} = B_{p,p-1} + \Phi_{p,p-1} B_{p-1,k-1}$

2. $C_{p,k-1} = C_{p,p-1} + \Phi_{p,p-1} C_{p-1,k-1}$

3. $\Gamma_{p,k-1} = \Gamma_{p,p-1} + \Phi_{p,p-1} \Gamma_{p-1,k-1}$

4. $c^{\Gamma}{}_{p,k-1} = c^{\Gamma}{}_{p,p-1} + \Phi_{p,p-1}\, c^{\Gamma}{}_{p-1,k-1}$

5. $\underline{\sigma}_p = \underline{\sigma}_{p-1} + \gamma_p \hat{\underline{u}}_{k-1}$

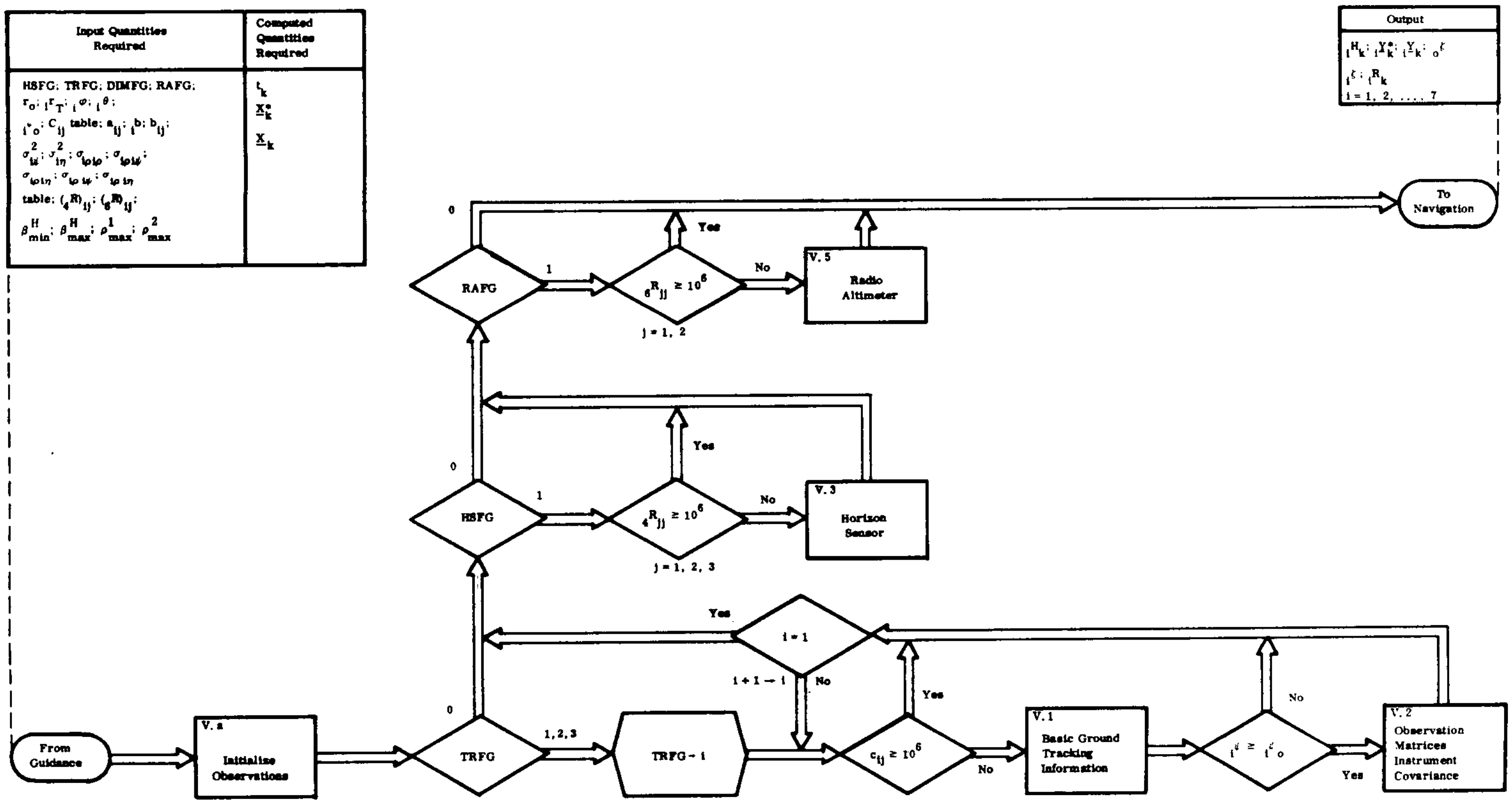
This block is entered when $t = t_p$ only. Some of the t_p points are t_k points in which case the output consists of $B_{k,k-1}$, $C_{k,k-1}$, $\Gamma_{k,k-1}$, $c^{\Gamma}{}_{k,k-1}$, and $\underline{\sigma}_k$. On the iteration following a t_k timepoint, $t_k \rightarrow t_{k-1}$ and $t = t_p \triangleq t_{k-1} + \Delta t_p$. At this time $t_{p-1} = t_{k-1}$ and $B_{p-1,k-1} = B_{k-1,k-1} = 0$. The comments made concerning B pertain to the C, Γ, and c^{Γ} also.

6. $\Phi_{p,k-1} = \Phi_{p,p-1} \Phi_{p-1,k-1}$

7. $c^{\Phi}{}_{p,k-1} = c^{\Phi}{}_{p,p-1}\; c^{\Phi}{}_{p-1,k-1}$

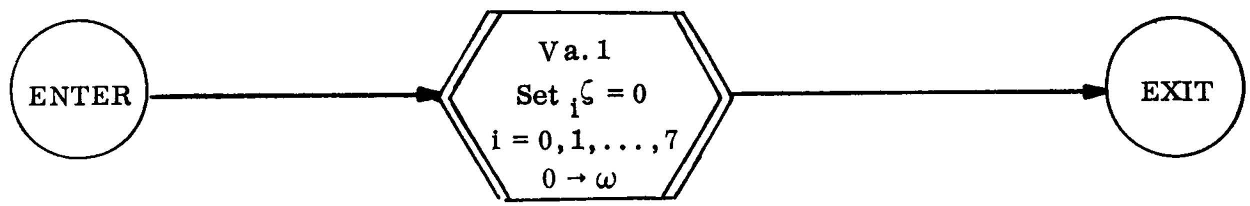
When $t_p = t_k$, $\Phi_{p,k-1} \triangleq \Phi_{k,k-1}$ and $c^{\Phi}{}_{p,k-1} \triangleq c^{\Phi}{}_{k,k-1}$. At the next entry, i.e., the next t_p time point, $\Phi_{p-1,k-1} \triangleq I$ and $c^{\Phi}{}_{p-1,k-1} \triangleq I$.

3.4.5 Electromagnetic Sensors



3.4.5.1 Level II Flow Chart - Electromagnetic Sensors

3.4.5.2 Detailed Flow Charts and Equations



3.4.5.2.1 **Initialize Observations - Block Va.**

V.1 Basic Ground Tracking Information

Range and range rate vector equations

Nominal

$${}_i\underline{\rho}^*_k = \underline{R}^*_k - {}_i\underline{r}_{Tk}$$

$${}_i\dot{\underline{\rho}}^*_k = \dot{\underline{R}}^*_k - {}_i\dot{\underline{r}}_{Tk}$$

Actual

$${}_i\underline{\rho}_k = \underline{R}_k - {}_i\underline{r}_{Tk}$$

$${}_i\dot{\underline{\rho}}_k = \dot{\underline{R}}_k - {}_i\dot{\underline{r}}_{Tk}$$

$\underline{R}^*_k$, $\dot{\underline{R}}^*_k$, $\underline{R}_k$, $\dot{\underline{R}}_k$ are the nominal and actual position and velocity, i.e.,

$$\underline{R}_k \stackrel{Df}{=} [X_{1k}, X_{2k}, X_{3k}]^T \equiv [X_k, Y_k, Z_k]^T$$

$$\dot{\underline{R}}_k \stackrel{Df}{=} [X_{4k}, X_{5k}, X_{6k}]^T \equiv [\dot{X}_k, \dot{Y}_k, \dot{Z}_k]^T$$

$${}_i\underline{r}_T = \begin{bmatrix} {}_iX_{Tk} \\ {}_iY_{Tk} \\ {}_iZ_{Tk} \end{bmatrix} = \begin{bmatrix} {}_ir_T \cos\varphi_i \cos({}_i\theta + \omega t_k) \\ {}_ir_T \cos\varphi_i \sin({}_i\theta + \omega t_k) \\ {}_ir_T \sin\varphi_i \end{bmatrix} \qquad i = 1, 2, 3$$

$${}_i\dot{\underline{r}}_T = \begin{bmatrix} -\omega_i Y_{Tk} \\ \omega_i X_{Tk} \\ 0 \end{bmatrix}$$

The equations which follow are identical in the nominal and actual trajectory. The nominal values will be designated by the superscript*.

Define inertial probe (w.r.t. tracker) positions

$$_i\underline{\rho} = \begin{bmatrix} {}_iX_p \\ {}_iY_p \\ {}_iZ_p \end{bmatrix}$$

Then the range and range rate from the trackers are

$$_i\rho = [{}_iX_p^2 + {}_iY_p^2 + {}_iZ_p^2]^{1/2}$$

$$_i\dot{\rho} = \frac{{}_i\underline{\rho}^T \cdot {}_i\dot{\underline{\rho}}}{{}_i\rho}$$

Elevation and azimuth angle of probe w.r.t. tracker

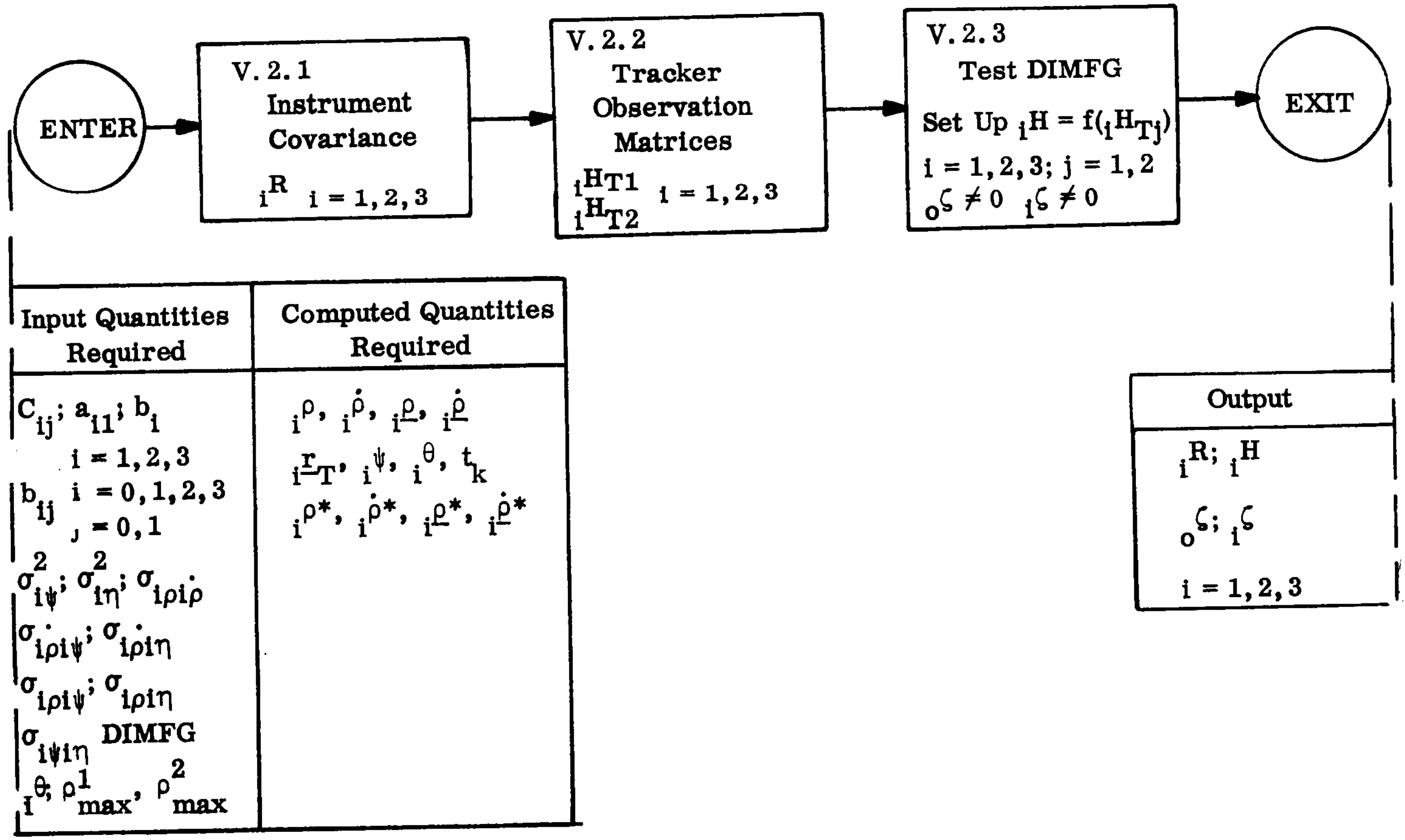
$$_i\psi = \sin^{-1}\left[\frac{{}_i\underline{r}_T^T \; {}_i\underline{\rho}}{{}_ir_T \; {}_i\rho}\right] \qquad -90^\circ \leq {}_i\psi \leq 90^\circ$$

$$_i\eta = \begin{cases} \cos^{-1}\left[\dfrac{-{}_ix_p''}{{}_i\rho \cos {}_i\psi}\right] \\ \\ \sin^{-1}\left[\dfrac{{}_iy_p''}{{}_i\rho \cos {}_i\psi}\right] \end{cases} \qquad 0 \leq {}_i\eta \leq 360^\circ$$

where

$$\begin{bmatrix} {}_ix_p'' \\ \\ {}_iy_p'' \end{bmatrix} = \begin{bmatrix} \sin {}_i\varphi \cos ({}_i\theta + \omega t_k) & \sin {}_i\varphi \sin ({}_i\theta + \omega t_k) & -\cos {}_i\varphi \\ \\ -\sin ({}_i\theta + \omega t_k) & \cos ({}_i\theta + \omega t_k) & 0 \end{bmatrix} \begin{bmatrix} {}_iX_p \\ {}_iY_p \\ {}_iZ_p \end{bmatrix}$$

$$
{}_i\underline{Y}^* \overset{\text{Df}}{=} \begin{bmatrix} {}_i\rho^* \\ {}_i\dot{\rho}^* \\ {}_i\psi^* \\ {}_i\eta^* \end{bmatrix} \quad ; \quad {}_i\underline{Y} \overset{\text{Df}}{=} \begin{bmatrix} {}_i\rho \\ {}_i\dot{\rho} \\ {}_i\psi \\ {}_i\eta \end{bmatrix} \qquad i = 1, 2, 3
$$



3.4.5.2.3 Observation Matrices and Instrument Covariances - Block V.2

V.2.1 Instrument Covariance

$$ {}_iR = {}_iC_j \begin{bmatrix} \sigma^2_{i\rho} & \sigma_{i\rho i\dot\rho} & \sigma_{i\rho i\psi} & \sigma_{i\rho i\eta} \\ & \sigma^2_{i\dot\rho} & \sigma_{i\dot\rho i\psi} & \sigma_{i\dot\rho i\eta} \\ & & \sigma^2_{i\psi} & \sigma_{i\psi i\eta} \\ \text{Symmetric} & & & \sigma^2_{i\eta} \end{bmatrix} \qquad i = 1, 2, 3 $$

Where C_{ij}'s are obtained from table look up.

Test ${}_i\rho \geq \rho^1_{max}$ if yes set $\sigma^2_{i\rho} = 10^6$

if no, compute $\sigma^2_{i\rho} = {}_ib_o + {}_ib_1\,{}_i\rho^2 + {}_ib_2\,{}_i\rho^4$

$$ \sigma^2_{i\dot\rho} = {}_ia_o + {}_ia_1\,(1 + {}_ib_i\dot\rho)^2\,{}_i\rho + {}_ia_2\,(1 + {}_ib_i\dot\rho)^2\,{}_i\rho^2 + {}_ia_3\,(1 + {}_ib_i\dot\rho)^4 $$

${}_i\rho$ and ${}_i\dot\rho$ are the actual values

Test ${}_i\rho \geq \rho^2_{max}$ if yes, set

$$ \left.\begin{matrix} \sigma^2_{i\psi} \\ \sigma^2_{i\eta} \end{matrix}\right\} = 10^6 $$

if no, use

$\sigma^2_{i\psi}$ and $\sigma^2_{i\eta}$ as in input.

V.2.2 Tracker Observation Matrices

$$
{}_i H_{T1} = \begin{bmatrix}
\frac{\partial_i \rho}{\partial X_1} & \frac{\partial_i \rho}{\partial X_2} & \frac{\partial_i \rho}{\partial X_3} & 0 & 0 & 0 \\
\frac{\partial_i \dot{\rho}}{\partial X_1} & \frac{\partial_i \dot{\rho}}{\partial X_2} & \frac{\partial_i \dot{\rho}}{\partial X_3} & \frac{\partial_i \dot{\rho}}{\partial X_4} & \frac{\partial_i \dot{\rho}}{\partial X_5} & \frac{\partial_i \dot{\rho}}{\partial X_6} \\
\frac{\partial_i \psi}{\partial X_1} & \frac{\partial_i \psi}{\partial X_2} & \frac{\partial_i \psi}{\partial X_3} & 0 & 0 & 0 \\
\frac{\partial_i \eta}{\partial X_1} & \frac{\partial_i \eta}{\partial X_2} & \frac{\partial_i \eta}{\partial X_3} & 0 & 0 & 0
\end{bmatrix} \qquad i = 1, 2, 3
$$

$$
{}_i H_{T2} = \begin{bmatrix}
\frac{\partial_i \rho}{\partial_i X_T} & \frac{\partial_i \rho}{\partial_i Y_T} & \frac{\partial_i \rho}{\partial_i Z_T} & 1 & 0 & 0 & 0 \\
\frac{\partial_i \dot{\rho}}{\partial_i X_T} & \frac{\partial_i \dot{\rho}}{\partial_i Y_T} & \frac{\partial_i \dot{\rho}}{\partial_i Z_T} & 0 & 1 & 0 & 0 \\
\frac{\partial_i \psi}{\partial_i X_T} & \frac{\partial_i \psi}{\partial_i Y_T} & \frac{\partial_i \psi}{\partial_i Z_T} & 0 & 0 & 1 & 0 \\
\frac{\partial_i \eta}{\partial_i X_T} & \frac{\partial_i \eta}{\partial_i Y_T} & \frac{\partial_i \eta}{\partial_i Z_T} & 0 & 0 & 0 & 1
\end{bmatrix} \qquad i = 1, 2, 3
$$

$$
\frac{\partial_i \rho}{\partial X_1} = \frac{\partial_i \dot{\rho}}{\partial X_4} = -\frac{\partial_i \rho}{\partial_i X_T} = \frac{{}_i X_{pk}}{{}_i \rho_k}
$$

$$
\frac{\partial_i \rho}{\partial X_2} = \frac{\partial_i \dot{\rho}}{\partial X_5} = -\frac{\partial_i \rho}{\partial_i Y_T} = \frac{{}_i Y_{pk}}{{}_i \rho_k}
$$

$$
\frac{\partial_i \rho}{\partial X_3} = \frac{\partial_i \dot{\rho}}{\partial X_6} = -\frac{\partial_i \rho}{\partial_i Z_T} = \frac{{}_i Z_{pk}}{{}_i \rho_k}
$$

$$\frac{\partial\, {}_i\dot{\rho}}{\partial X_1} = {}_i\dot{X}_{pk}\left[\frac{1}{{}_i\rho_k} - \frac{{}_iX_{pk}^2}{{}_i\rho_k^3}\right] - {}_i\dot{Y}_{pk}\,\frac{{}_iX_{pk}\,{}_iY_{pk}}{{}_i\rho_k^3} - {}_i\dot{Z}_{pk}\,\frac{{}_iX_{pk}\,{}_iZ_{pk}}{{}_i\rho_k^3}$$

$$\frac{\partial\, {}_i\dot{\rho}}{\partial X_2} = -\,{}_i\dot{X}_{pk}\,\frac{{}_iX_{pk}\,{}_iY_{pk}}{{}_i\rho_k^3} + {}_i\dot{Y}_{pk}\left[\frac{1}{{}_i\rho_k} - \frac{{}_iY_{pk}^2}{{}_i\rho_k^3}\right] - {}_i\dot{Z}_{pk}\,\frac{{}_iY_{pk}\,{}_iZ_{pk}}{{}_i\rho_k^3}$$

$$\frac{\partial\, {}_i\dot{\rho}}{\partial X_3} = -\,{}_i\dot{X}_{pk}\,\frac{{}_iX_{pk}\,{}_iZ_{pk}}{{}_i\rho_k^3} - {}_i\dot{Y}_{pk}\,\frac{{}_iY_{pk}\,{}_iZ_{pk}}{{}_i\rho_k^3} + {}_i\dot{Z}_{pk}\left[\frac{1}{{}_i\rho_k} - \frac{{}_iZ_{pk}^2}{{}_i\rho_k^3}\right]$$

$$\frac{\partial\, {}_i\dot{\rho}}{\partial\, {}_iX_T} = -\frac{\partial\, {}_i\dot{\rho}}{\partial X_1} - \frac{\omega\, {}_iY_p}{{}_i\rho} \qquad \frac{\partial\, {}_i\dot{\rho}}{\partial\, {}_iY_T} = -\frac{\partial\, {}_i\dot{\rho}}{\partial X_2} + \frac{\omega\, {}_iX_p}{{}_i\rho} \qquad \frac{\partial\, {}_i\dot{\rho}}{\partial\, {}_iZ_T} = -\frac{\partial\, {}_i\dot{\rho}}{\partial X_3}$$

$$\frac{\partial\, {}_i\psi}{\partial X_1} = \frac{1}{\cos\, {}_i\psi}\left[\frac{{}_iX_T}{{}_ir_T\,{}_i\rho} - \frac{{}_iX_p}{{}_i\rho^2}\,\sin\, {}_i\psi\right]$$

$$\frac{\partial\, {}_i\psi}{\partial X_2} = \frac{1}{\cos\, {}_i\psi}\left[\frac{{}_iY_T}{{}_ir_T\,{}_i\rho} - \frac{{}_iY_p}{{}_i\rho^2}\,\sin\, {}_i\psi\right]$$

$$\frac{\partial\, {}_i\psi}{\partial X_3} = \frac{1}{\cos\, {}_i\psi}\left[\frac{{}_iZ_T}{{}_ir_T\,{}_i\rho} - \frac{{}_iZ_p}{{}_i\rho^2}\,\sin\, {}_i\psi\right]$$

$$\frac{\partial\, {}_i\psi}{\partial\, {}_iX_T} = \frac{1}{\cos\, {}_i\psi}\left[\frac{{}_iX_p}{{}_ir_T\,{}_i\rho} - \frac{{}_iX_T}{{}_ir_T^2}\,\sin\, {}_i\psi\right] - \frac{\partial\, {}_i\psi}{\partial X_1}$$

$$\frac{\partial\, {}_i\psi}{\partial\, {}_iY_T} = \frac{1}{\cos\, {}_i\psi}\left[\frac{{}_iY_p}{{}_ir_T\,{}_i\rho} - \frac{{}_iY_T}{{}_ir_T^2}\,\sin\, {}_i\psi\right] - \frac{\partial\, {}_i\psi}{\partial X_2}$$

$$\frac{\partial\, {}_i\psi}{\partial\, {}_iZ_T} = \frac{1}{\cos\, {}_i\psi}\left[\frac{{}_iZ_p}{{}_ir_T\,{}_i\rho} - \frac{{}_iZ_T}{{}_ir_T^2}\,\sin\, {}_i\psi\right] - \frac{\partial\, {}_i\psi}{\partial X_3}$$

$$\frac{\partial_i \eta}{\partial X_1} = \frac{1}{{}_i x''_p} \left[+ \sin ({}_i\theta + \omega t_k) + \frac{{}_i y''_p}{{}_i \rho} \left(\frac{\partial_i \rho}{\partial X_1} \right) - {}_i y''_p \left(\frac{\sin {}_i \psi}{\cos {}_i \psi} \right) \frac{\partial_i \psi}{\partial X_1} \right]$$

$$\frac{\partial_i \eta}{\partial X_2} = \frac{1}{{}_i x''_p} \left[- \cos ({}_i\theta + \omega t_k) + \frac{{}_i y''_p}{{}_i \rho} \left(\frac{\partial_i \rho}{\partial X_2} \right) - {}_i y''_p \left(\frac{\sin {}_i \psi}{\cos {}_i \psi} \right) \frac{\partial_i \psi}{\partial X_2} \right]$$

$$\frac{\partial_i \eta}{\partial X_3} = \frac{1}{{}_i x''_p} \left[\frac{{}_i y''_p}{{}_i \rho} \left(\frac{\partial_i \rho}{\partial X_3} \right) - {}_i y''_p \left(\frac{\sin {}_i \psi}{\cos {}_i \psi} \right) \left(\frac{\partial_i \psi}{\partial X_3} \right) \right]$$

$$\frac{\partial_i \eta}{\partial_i X_T} = \frac{1}{{}_i x''_p} \left[-\sin ({}_i\theta + \omega t_k) \frac{-A\,{}_i Y_T}{{}_i X_T^{\,2} + {}_i Y_T^{\,2}} + \frac{{}_i y''_p}{{}_i \rho} \left(\frac{\partial_i \rho}{\partial_i X_T} \right) - {}_i y''_p \left(\frac{\sin {}_i \psi}{\cos {}_i \psi} \right) \frac{\partial_i \psi}{\partial_i X_T} \right]$$

$$\frac{\partial_i \eta}{\partial_i Y_T} = \frac{1}{{}_i x''_p} \left[\cos ({}_i\theta + \omega t_k) + \frac{A\,{}_i X_T}{{}_i X_T^2 + {}_i Y_T^2} + \frac{{}_i y''_p}{{}_i \rho} \left(\frac{\partial_i \rho}{\partial_i Y_T} \right) - {}_i y''_p \left(\frac{\sin {}_i \psi}{\cos {}_i \psi} \right) \frac{\partial_i \psi}{\partial_i Y_T} \right]$$

$$\frac{\partial_i \eta}{\partial_i Z_T} = \frac{1}{{}_i x''_p} \left[\frac{{}_i y''_p}{{}_i \rho} \left(\frac{\partial_i \rho}{\partial_i Z_T} \right) - {}_i y''_p \left(\frac{\sin {}_i \psi}{\cos {}_i \psi} \right) \left(\frac{\partial_i \psi}{\partial_i Z_T} \right) \right]$$

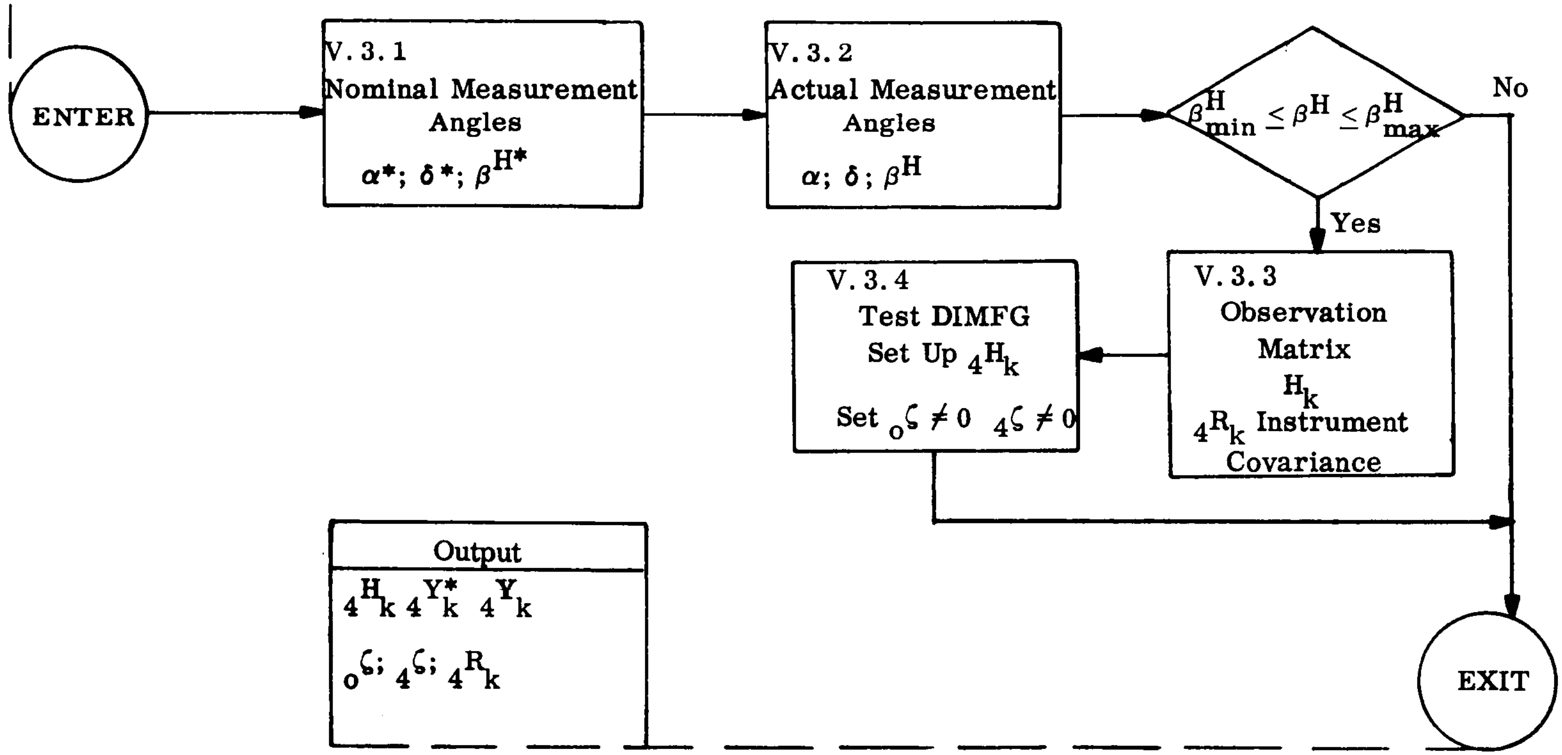
$$A = {}_i X_\rho \cos ({}_i\theta + \omega t) + {}_i Y_\rho \sin ({}_i\theta + \omega t)$$

These partials are evaluated employing the nominal (*) values.

V.2.3 Set Up of Observation Matrix

The observation matrix form is set up in Dimension Block B.5. It remains here to place the computed submatrice in the proper locations, and set ${}_o\zeta \neq 0$ ${}_i\zeta \neq 0$ $i = 1, 2, 3$.

Input Quantities Required	Computed Quantities Required
DIMFG	$\underline{R}^*_k$
r_o	$\underline{R}_k$
$({}_4R)_{ij}$ Table	
β^H_{min} ; β^H_{max}	



3.4.5.2.4 Horizon Sensor - Block V.3

V. 3. 1 Nominal Measurement Angles

Three angles are measured by this instrument

$$\text{Elevation angle} \quad \alpha^* = -\sin^{-1}\left(\frac{X^*_{3k}}{R^*_k}\right)$$

$$\text{Azimuth angle} \quad \delta^* = \begin{cases} \sin^{-1}\left(\dfrac{X^*_{2k}}{[X^{*2}_{1k} + X^{*2}_{2k}]^{1/2}}\right) \\ \cos^{-1}\left(\dfrac{X^*_{1k}}{[X^{*2}_{1k} + X^{*2}_{2k}]^{1/2}}\right) \end{cases}$$

$$\text{Subtended angle} \quad \beta^{H*} = \sin^{-1}\left(\frac{r_o}{R^*_k}\right)$$

where r_o = radius of the planet

$$_4\underline{Y}^*_k \stackrel{Df}{=} \begin{bmatrix} \alpha^* \\ \delta^* \\ \beta^{H*} \end{bmatrix}$$

V. 3. 2 Actual Measurement Angles

$$_4Y \stackrel{Df}{=} \begin{bmatrix} \alpha \\ \delta \\ \beta^H \end{bmatrix}$$

The equations are the same as in V. 3. 1 except that $\underline{R}_k \rightarrow \underline{R}^*_k$.

Test $\beta^H_{min} \leq \beta^H \leq \beta^H_{max}$; if yes, continue; if no, exit.

V. 3. 3 Observation Matrix and Instrument Covariance

$$H_k = (\underset{3 \times 3}{H} \quad \underset{3 \times 3}{O})$$

The nontrivial portion of the H_k matrix has dimension (3 x 3). It is represented by the following matrix

$$\begin{bmatrix} \dfrac{-\sin \alpha^* \cos \delta^*}{R^*_k} & \dfrac{-\sin \alpha^* \sin \delta^*}{R^*_k} & \dfrac{-X^*_{2k}}{R^{*2}_k \sin \delta^*} \\ \dfrac{-\sin^2 \delta^*}{X^*_{2k}} & \dfrac{\cos^2 \delta^*}{X^*_{1k}} & 0 \\ \dfrac{-X^*_{1k} \tan \beta^{H^*}}{R^{*2}_k} & \dfrac{-X^*_{2k} \tan \beta^{H^*}}{R^{*2}_k} & \dfrac{-X^*_{3k} \tan \beta^{H^*}}{R^{*2}_k} \end{bmatrix}$$

Note: $\tan\beta^{H^*} = \dfrac{r_o}{\sqrt{R^{*2}_k - r^2_o}}$

The instrument covariance is given by

$$_4R_k = R_{ij} \qquad \begin{matrix} i = 1,2,3 \\ j = 1,2,3 \end{matrix}$$

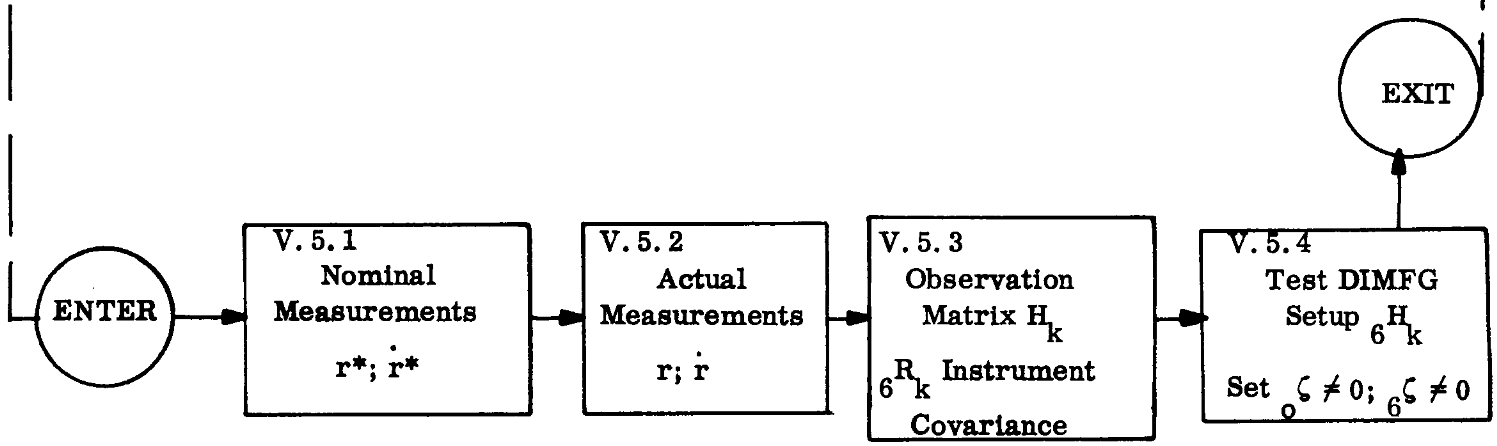
where the R_{ij}'s are obtained from a table look up as a function of time. Since $_4R$ is symmetric, only the non-symmetric elements will be part of the input.

V.3.4 Setup of $_4H_k$

The observation matrix form is established in Block B.5. It remains here to place the computed submatrices in the proper locations. After $_4H_k$ is set up, $_o\zeta$ and $_4\zeta$ are set $\neq 0$.

Input Quantities Required	Computed Quantities Required
DIMFG; r_o; $({}_6R)_{ij}$ table	$\underline{X}^*_k$ $\underline{X}_k$

Output
${}_6H_k$; ${}_6Y^*_k$; ${}_6Y_k$; ${}_6R_k$; ${}_o\zeta$; ${}_6\zeta$



3.4.5.2.6 Radio Altimeter - Block V.5

V.5.1 Nominal Measurements

Radial altitude $r_k^* = R_k^* - r_o$

Radial speed $\dot{r}_k^* = \dfrac{\underline{R}_k^* \cdot \dot{\underline{R}}_k^*}{R_k^*}$

$$_6\underline{Y}_k^* \stackrel{Df}{=} \begin{bmatrix} r^* \\ \dot{r}^* \end{bmatrix}$$

V.5.2 Actual Measurements

$$_6\underline{Y}_k \stackrel{Df}{=} \begin{bmatrix} r \\ \dot{r} \end{bmatrix}$$

The equations are the same as in V.5.1 except $\underline{R}_k \to \underline{R}_k^*$; $\dot{\underline{R}}_k \to \dot{\underline{R}}_k^*$, etc.

V.5.3 Observation Matrix and Instrument Covariance

$$H_k = [\underset{2 \times 6}{{}_R H}, \underset{2 \times 2}{{}_{RB} H}]$$

$${}_R H = \begin{bmatrix} \dfrac{\partial r}{\partial X_1} & \dfrac{\partial r}{\partial X_2} & \dfrac{\partial r}{\partial X_3} & 0 & 0 & 0 \\ \dfrac{\partial \dot{r}}{\partial X_1} & \dfrac{\partial \dot{r}}{\partial X_2} & \dfrac{\partial \dot{r}}{\partial X_3} & \dfrac{\partial \dot{r}}{\partial X_4} & \dfrac{\partial \dot{r}}{\partial X_5} & \dfrac{\partial \dot{r}}{\partial X_6} \end{bmatrix}$$

$${}_{RB} H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{when BSFG} \neq 0$$

$$\frac{\partial r}{\partial X_1} = \frac{\partial \dot{r}}{X_4} = \frac{X^*_{1k}}{R_k^*}$$

$$\frac{\partial r}{\partial X_2} = \frac{\partial \dot{r}}{\partial X_5} = \frac{X^*_{2k}}{R^*_k}$$

$$\frac{\partial r}{\partial X_3} = \frac{\partial \dot{r}}{\partial X_6} = \frac{X^*_{3k}}{R^*_k}$$

$$\frac{\partial \dot{r}}{\partial X_1} = \frac{1}{R^*_k}\left[X^*_{4k} - \dot{r}^*_k \frac{X^*_{1k}}{R^*_k}\right]$$

$$\frac{\partial \dot{r}}{\partial X_2} = \frac{1}{R^*_k}\left[X^*_{5k} - \dot{r}^*_k \frac{X^*_{2k}}{R^*_k}\right]$$

$$\frac{\partial \dot{r}}{\partial X_3} = \frac{1}{R^*_k}\left[X^*_{6k} - \dot{r}^*_k \frac{X^*_{3k}}{R^*_k}\right]$$

The instrument covariance is given by

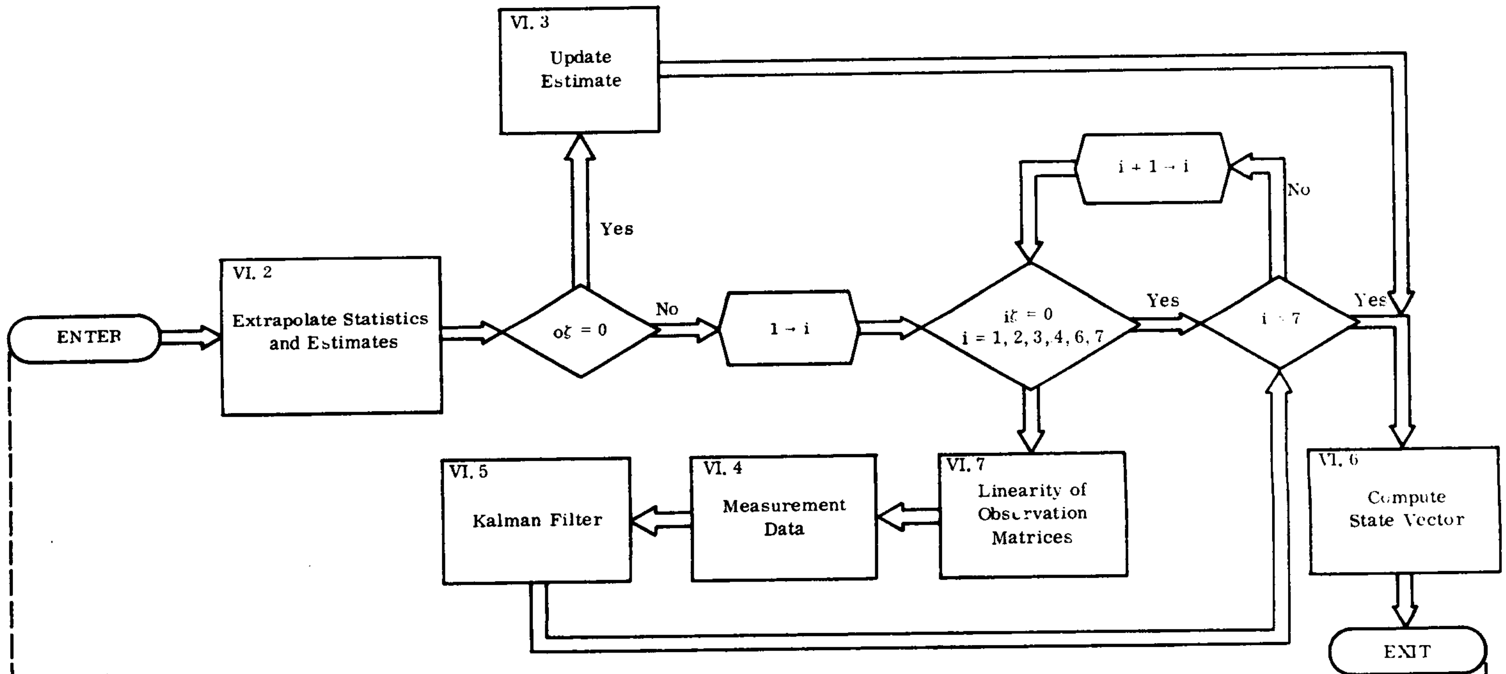
$$_6R_k = \begin{bmatrix} R_{11} & R_{12} \\ R_{12} & R_{22} \end{bmatrix}$$

where R_{11}, R_{22}, R_{12} are obtained from a table look up as a function of time.

V.5.4 Setup of $_6H_k$

The observation matrix form is established in Block B.6. It remains here to place the computed submatrices in the proper locations. After $_6H_k$ is set up, $_0\zeta$ and $_6\zeta$ are set $\neq 0$.

3.4.6 Navigation



Stored	Computed
$_i\sigma$ (i = 1, 2, 3, 4, 6, 7)	$\Phi_{k,k-1}$ (6×6), $\Gamma_{k,k-1}$ (6×2), $c\Gamma_{k,k-1}$ (1×1)
$_1Q_k$ table, IMFG	γ_k (3×2), $B_{k,k-1}$ (6×2), $C_{k,k-1}$ (6×1)
k_0, k_1, k_2, k_3, h_0, $\dot{h}$, h	$c\Phi_{k,k-1}$ (1×1), $\hat{\underline{u}}_{k-1}$ (2×1), IMFG,
$_iR_0$, $_iR$ (i = 1, 2, 3, 4, 6, 7)	k_0, k_1, k_2, k_3, $\dot{h}$, h, h_0,
	$_i\underline{\alpha}$ (i = 4, 6), $_i\underline{d}$ (7×1) (i = 1, 2, 3),
	$\underline{\epsilon}$ (15×1), $_i\underline{Y}_k$ (var×1), $_iY_k^*$ (var×1)
	$_iH_k$ (var×var), $\underline{\sigma}_k$ (3×1),
	$\int_{t_0}^{t} G\,dt$ (3×15), $\underline{x}_{Dif}$, $\underline{b}'$ (2×1)
	$\underline{c}_k$ (1×1), $\underline{u}_{k-1}$, γ_k (3×2)

Output
$a\hat{\underline{x}}_k$ (9 × 1)
Q_k (3 × 3)
$_1Q_{oo}$ (1 × 1)

3.4.6.1 Level II Flow Chart - Navigation

3.4.6.2 Detailed Flow Charts and Equations

3.4.6.2.2 Extrapolate Statistics and Estimate - Block VI.2

Input: $\Phi_{k,k-1})$ (6x6), ${}_A P_{k-1}$ (nxn), $\Gamma_{k,k-1}$ (6x2), ${}_c\Gamma_{k,k-1}$ (1x1), γ_k (3x2), ${}_1Q_k$ Table, $B_{k,k-1}$ (6x2), $C_{k,k-1}$ (6x1), ${}_c\Phi_{k,k-1}$ (1x1), ${}_A\hat{\underline{x}}_{k-1}$ (nx1), $\hat{\underline{u}}_{k-1}$ (2x1), IMFG, k_o, k_1, k_2, k_3, $\dot{h}$, h, h_o

Output: ${}_A P'_k$ (nxn), ${}_A\hat{\underline{x}}'_k$ (nx1), Q_k (3x3)

1. $${}_2Q_k = |\dot{h}| \left(k_o + [k_1 + k_2 h] e^{-k_3 [h - h_o]}\right)$$

2. Q_k is a 2x2 symmetric matrix which is obtained from table lookup as described in the NAVIGATION section of 3.1.

3. Form Q_k which looks like

$$Q_k \triangleq \begin{bmatrix} {}_1Q_k & 0 \\ 0 & {}_2Q_k \end{bmatrix}$$

4. Form ${}_A\Delta_{k,k-1}$

$${}_A\Delta_{k,k-1} = \begin{bmatrix} \Gamma_{k,k-1} & {}_1 0 \\ {}_2 0 & {}_3 0 \\ {}_4 0 & {}_c\Gamma_{k,k-1} \\ \gamma_k & {}_6 0 \\ {}_5 0 & {}_7 0 \end{bmatrix}$$

γ_k and ${}_6 0$ are used only if IMFG $\neq$ 0. The dimensions of ${}_A\Delta_{k,k-1}$ and the partitioned 0 matrices are given below.

Dim $[{}_1 0]$ is (6x1) Dim $[{}_4 0]$ is (1x2) Dim $[{}_A\Delta_{k,k-1}]$ is (nx3)
Dim $[{}_2 0]$ is (2x2) Dim $[{}_5 0]$ is (n-12x2) Dim $[{}_7 0]$ is (n-12x1)
Dim $[{}_3 0]$ is (2x1) Dim $[{}_6 0]$ is (3x1)

5. Form ${}_A\Phi_{k,k-1}$

$$ {}_A\Phi_{k,k-1} = \left[\begin{array}{ccc:c} \Phi_{k,k-1} & B_{k,k-1} & C_{k,k-1} & \\ {}_8 0 & {}_1 I & {}_9 0 & {}_{12} 0 \\ {}_{10} 0 & {}_{11} 0 & c\Phi_{k,k01} & \\ \hdashline & {}_{13} 0 & & {}_2 I \end{array}\right] $$

The dimensions of ${}_A\Phi_{k,k-1}$ and its partitioned submatrices is given below.

Dim $[\Phi_{k,k-1}]$ is (6x6) Dim $[{}_9 0]$ is (2x1) Dim $[{}_{13} 0]$ is (n-9x9)
Dim $[B_{k,k-1}]$ is (6x2) Dim $[{}_{10} 0]$ is (1x6) Dim $[{}_1 I]$ is (2x2)
Dim $[C_{k,k-1}]$ is (6x1) Dim $[{}_{11} 0]$ is (1x2) Dim $[{}_2 I]$ is (n-9xn-9)
Dim $[{}_8 0]$ is (2x6) Dim $[{}_{12} 0]$ is (2xn-9) Dim $[{}_A\Phi_{k,k-1}]$ is (nxn)

6. $$ {}_A P'_k = [{}_A\Phi_{k,k-1}][{}_A P_{k-1}][{}_A\Phi^T_{k,k-1}] + [{}_A\Delta_{k,k-1}][Q_{k-1}][{}_A\Delta^T_{k,k-1}] $$

7. $$ {}_A\Gamma_{k,k-1} \triangleq \begin{bmatrix} \Gamma_{k,k-1} \\ {}_{14} 0 \end{bmatrix} ; \qquad \text{Dim } [{}_{14} 0] \text{ is (n-6x2)} $$

8. $$ {}_A\underline{\hat{x}}'_k = {}_A\Phi_{k,k-1}\, {}_A\underline{\hat{x}}_{k-1} + {}_A\Gamma_{k,k-1}\, \underline{\hat{u}}_{k-1} $$

3.4.6.2.3 Update Estimate - Block VI. 3

Input: ${}_{A}\hat{\underline{x}}'_{k}$ (nx1), ${}_{A}P'_{k}$ (nxn), ${}_{A}\Phi_{k,k-1}$ (nxn)

Output: ${}_{i}K_{k}$ (n x var), ${}_{A}\hat{\underline{x}}_{k}$ (nx1), ${}_{A}P_{k}$ (nxn), ${}_{i}\underline{z}_{k}$ (var x var)

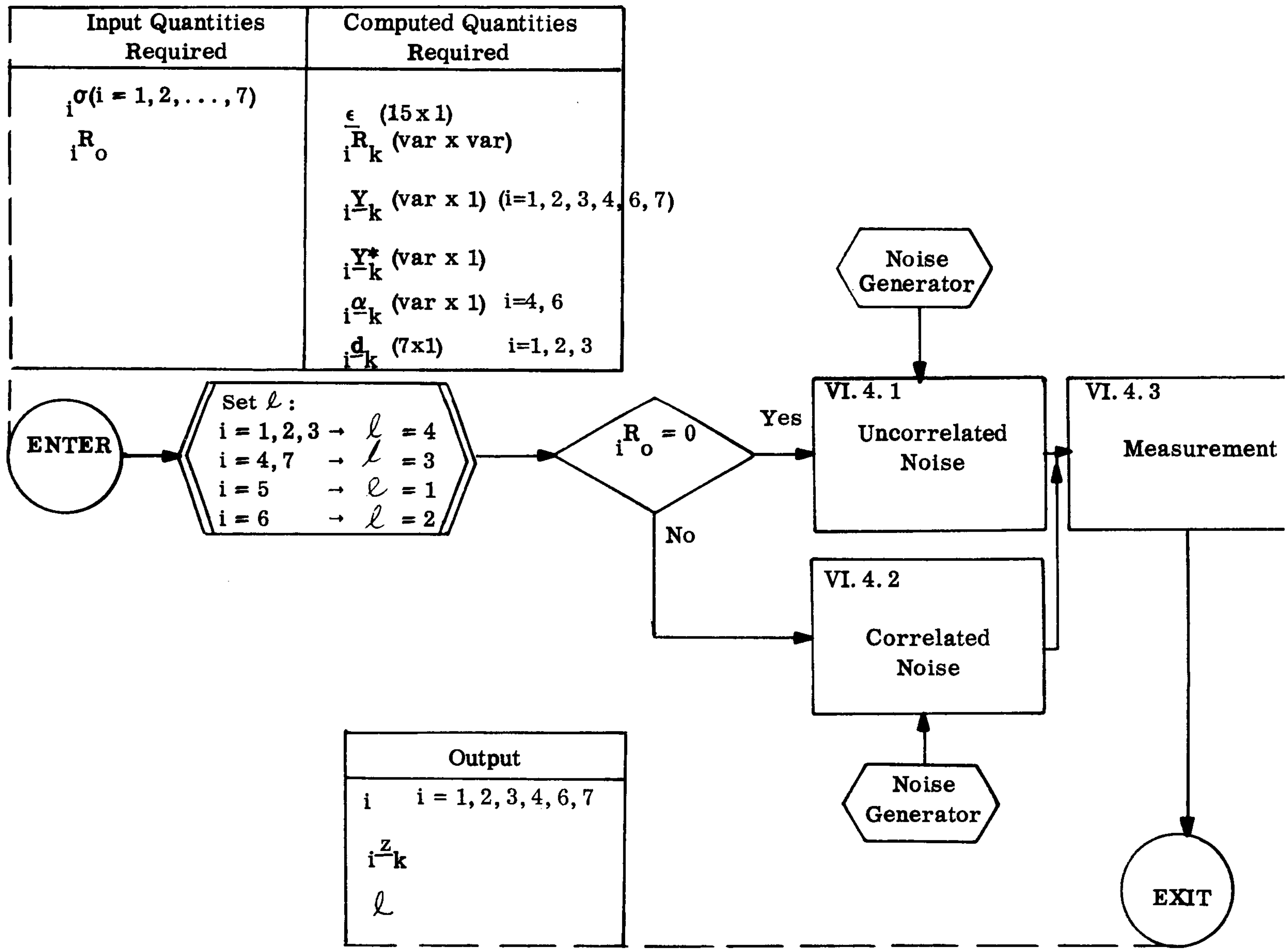
1. ${}_{i}K_{k} \equiv 0$ $\qquad i = 1, 2, \ldots, 7$

2. ${}_{A}\hat{\underline{x}}_{k} \equiv {}_{A}\hat{\underline{x}}'_{k}$

3. ${}_{i}\underline{z}_{k} \equiv 0$ $\qquad i = 1, 2, \ldots, 7$

4. ${}_{A}P_{k} \equiv {}_{A}P'_{k}$

Note: This block is not coded in the program, but the effect described herein is obtained by storing the computed quantities in equations 1 through 4 in the same locations as these quantities with primes affixed to them. Thus the gains, ${}_{i}K_{k}$, the estimate of the state, ${}_{A}\hat{\underline{x}}_{k}$, the measurements, ${}_{i}\underline{z}_{k}$, and the covariance, ${}_{A}P_{k}$, before measurements by the sensors have the same FORTRAN symbol as the extrapolated values.



3.4.6.2.4 Measurement Data - Block VI. 4

Block VI. 4. 1 Uncorrelated Noise

Input: ℓ, ${}_i\sigma$, ${}_iR_k$ (var x var)

Output: ${}_i\underline{v}_k$ (var x 1)

Generate gaussian random numbers with mean zero and variance determined by the diagonal elements of the ${}_iR_k$ matrix under the rule that:

variance of the j^{th} random number = $(1 + {}_i\sigma)\,{}_ir_{jj}(t_k)$

$j = 1, 2, \ldots$ where $r_{jj}(t_k)$ is a diagonal element of ${}_iR_k$

These random numbers shall form the vector

$$
{}_i\underline{v}_k = \begin{bmatrix} {}_iv'_1 \\ {}_iv'_2 \\ \vdots \\ {}_iv'_\ell \end{bmatrix}
$$

Block VI. 4. 2 Correlated Noise

Input: ℓ, ${}_i\sigma$, ${}_iR_k$ ($\ell \times \ell$)

Output: ${}_i\underline{v}_k$ (var x 1)

1. ${}_iR_k$ is sent to the Triangularization Subroutine. Output of this routine is a diagonal matrix ${}_iD_R$ (var x var) and a lower triangular matrix ${}_iT_Q$ (var x var).

2. Using the noise generator and the diagonal elements of ${}_iD_R$, generate ℓ gaussian random numbers. The variances of the ℓ random numbers are (1 + ${}_i\sigma$) times the individual diagonal elements of the diagonal matrix.

3. Pre-multiply the vector of ℓ elements by ${}_iT_Q$ to give ${}_i\underline{v}_k$.

Block VI. 4. 3 Measurements

Input: ${}_i\underline{Y}_k$ (var x 1), ${}_i\underline{Y}_k^*$ (var x 1), ${}_i\underline{v}_k$ (var x 1) (i = 1, 2, 3, 4, 6, 7)

${}_iH_{T2}$ (4x7) (i = 1, 2, 3), ${}_i\underline{\alpha}$ (var x 1) (i = 4, 6)

${}_i\underline{d}$ (7x1) (i = 1, 2, 3), $\underline{\sigma}_k$ (3x1), $\underline{\epsilon}$ (15 x 1), $\int_{t_i}^{t_k} G\,dt$ (3x15)

Output: ${}_i\underline{z}_k$ (i = 1, 2, 3, 4, 6, 7)

1. $${}_i\underline{z}_k = {}_i\underline{Y}_k - {}_i\underline{Y}^* + [{}_iH_{T2}][{}_i\underline{d}] + {}_i\underline{v}_k \qquad (i=1, 2, 3)$$

2. $${}_i\underline{z}_k = {}_i\underline{Y}_k - {}_i\underline{Y}^* + {}_i\underline{\alpha} + {}_i\underline{v}_k \qquad (i=4, 6)$$

3. $${}_7\underline{z}_k = {}_7\underline{Y}_k - {}_7\underline{Y}_k^* + \int_{t_o}^{t_k} G\,dt\,\underline{\epsilon} + {}_7\underline{v}_k - \underline{\sigma}_k$$

3.4.6.2.5 Kalman Filter - Block VI.5

Input: ${}_A\hat{\underline{x}}'_k(n\times 1)$, ${}_AP'_k(n\times n)$, ${}_iH_k(\text{var}\times n)$, ${}_iR_k(\text{var}\times\text{var})$, ${}_i\underline{z}_k(\text{var}\times 1)(i=1,2,3,4,6,7)$

Output: ${}_A\hat{\underline{x}}_k(n\times 1)$, ${}_AP_k(n\times n)$

1. Is this first entry at current t_k time point?

   a. Yes:

$${}_A\hat{\underline{x}}''_k \triangleq {}_A\hat{\underline{x}}'_k$$

$${}_AP''_k \triangleq {}_AP'_k$$

   b. No:

$${}_A\hat{\underline{x}}''_k \triangleq {}_A\hat{\underline{x}}_k$$

$${}_AP''_k \triangleq {}_AP_k$$

2. $${}_iK_k = [{}_AP''_k][{}_iH_k^T]\{[{}_iH_k][{}_AP''_k][{}_iH_k^7] + [{}_iR_k]\}^{-1}$$

3. $${}_A\hat{\underline{x}}_k = {}_A\underline{x}''_k + {}_iK_k\{{}_i\underline{z}_k - {}_iH_k\,{}_A\hat{\underline{x}}''_k\}$$

4. $${}_AP_k = (I - [{}_iK_k][{}_iH_k])\,{}_AP''_k\,(I - [{}_iK_k][{}_iH_k])^T + [{}_iK_k][{}_iR_k][{}_iK_k]^T$$

3.4.6.2.6 Compute State Vector - Block VI. 6

Input: $\underline{x}_{Dif}$(6x1), $\hat{\underline{x}}_k$, $\Phi_{k,k-1}$(6x6), $B_{k,k-1}$(6x2), $C_{k,k-1}$(6x1), $\Gamma_{k,k-1}$(6x2), $\underline{x}_{k-1}$(6x1), $\underline{b}'$(2x1), $\underline{c}_k$(1x1), $\underline{u}_{k-1}$ (2x1), $\underline{x}_o$ (6x1)

Output: $\tilde{\underline{x}}_k$ (6x1), $\underline{x}_k$(6x1)

1. $\tilde{\underline{x}}_k = \underline{x}_{Dif} - \hat{\underline{x}}_k$

2. $\underline{x}_k = \Phi_{k,k-1}\underline{x}_{k-1} + B_{k,k-1}\underline{b}' + C_{k,k-1}\underline{c}_{k-1} + \Gamma_{k,k-1}\underline{u}_{k-1}$

3.4.6.2.7 Linearity of Observation Matrices - Block VI.7

Input: $\gamma_k (3x2)$, $\underline{w}_{k-1} (2x1)$, $\underline{b}' (2x1)$, $\underline{c}_k (1x1)$

Output: ${}_i\Delta y$ (i=1, 2, 3, 4, 6, 7)

1. Is i = 7?

a. No: $${}_i\Delta\underline{y} = ({}_i\underline{Y} - {}_i\underline{Y}^*) - {}_{ik}H_k \, \underline{x}_{Dif}$$

where ${}_{ik}H_k \triangleq {}_1H_{T1}, {}_2H_{T1}, {}_3H_{T1}, {}_HH, {}_RH$ when i = 1, 2, 3, 4, 6 respectively

b. Yes:

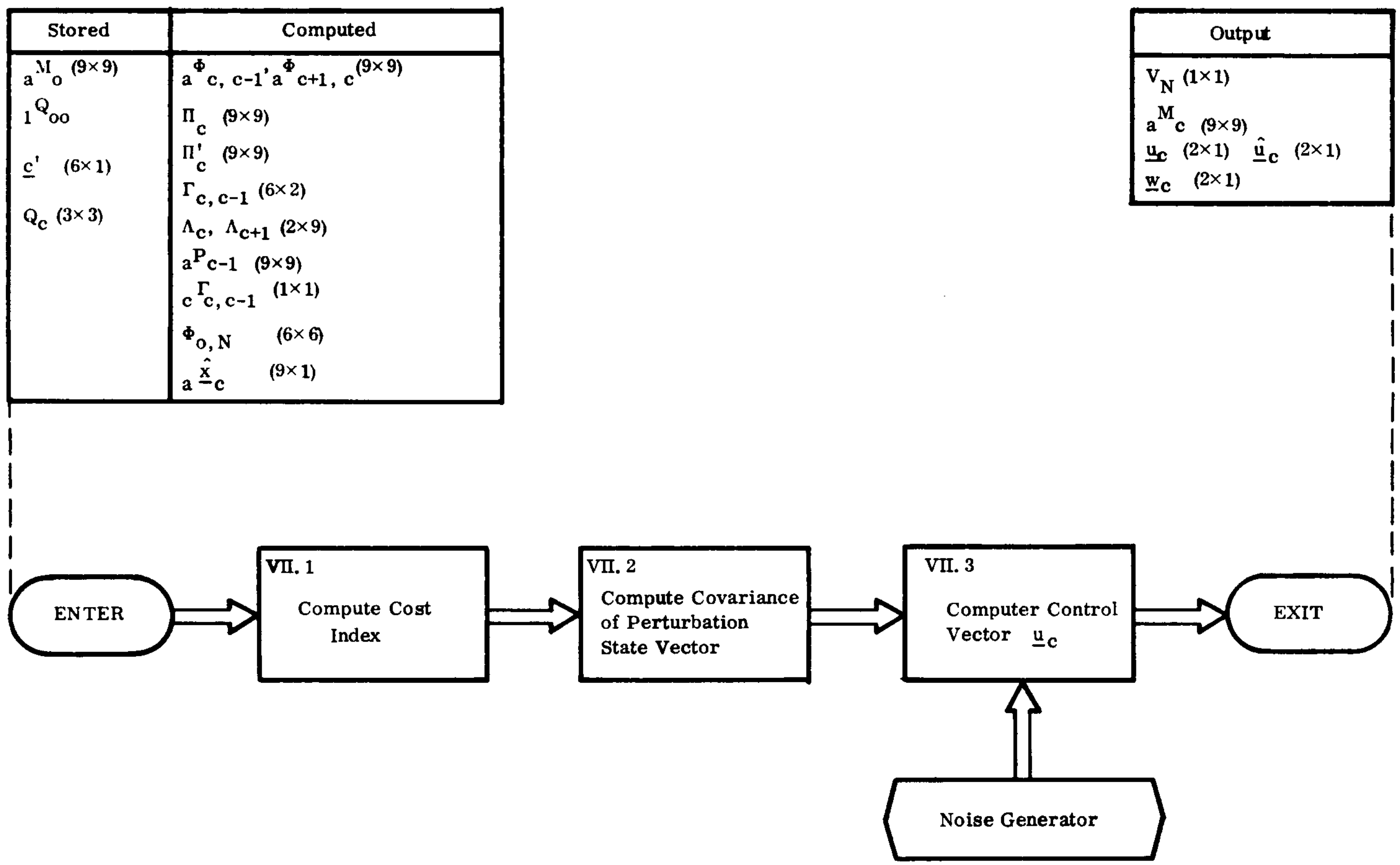
$$\underline{\eta}_k = \underline{\eta}_{k-1} + \gamma_k \underline{w}_{k-1}$$

$${}_{aa}\underline{x}^T_{Dif} \triangleq [\underline{x}^T_{Dif} (1x6) \;\; \underline{b}'^T (1x2) \;\; \underline{c}_{k-1} (1x1) \;\; \underline{\eta}^T_k (1x3)]$$

$${}_{7a}H_k \triangleq [{}_aJ_k (3x6) \;\; {}_2J_k (3x2) \;\; {}_3J_k (3x1) \;\; I\ (3x3)]$$

$${}_7\Delta\underline{y} = {}_7\underline{Y} - \underline{Y}^* - \underline{\sigma}_k - [{}_{7a}H_k][{}_{aa}\underline{x}_{Dif}]$$

3.4.7 Guidance



3.4.7.1 Level II Flow Chart - Guidance

3.4.7.2 Detailed Flow Charts and Equations

3.4.7.2.1 Compute Cost Index - Block VII. 1

INPUT: ${}_a\Phi_{c,c-1}$ (9x9), π_c (9x9), ${}_aM_o$ (9x9), $\pi'_{c'}$ (9x9), $\Gamma_{c,c-1}$ (6x2), Λ_c (2x9), ${}_aP_{c-1}$ (9x9), Q_c (3x3), ${}_c\Gamma_{c,c-1}$(1x1)

OUTPUT: V_N (1x1), $\Delta_{c,c-1}$(9x3), Q_{c-1} (3x3)

This output is required when $t = t_N$

1) $${}_a\Gamma_{c,c-1} = \begin{bmatrix} \Gamma_{c,c-1} \\ {}_1 0 \end{bmatrix}$$

$$\Delta_{c,c-1} = \begin{bmatrix} \Gamma_{c,c-1} & {}_2 0 \\ {}_3 0 & {}_4 0 \\ {}_5 0 & {}_c\Gamma_{c,c-1} \end{bmatrix}$$

3) $$V_N = \text{trace}\left\{\left[{}_a\Phi_{1,0}^T\right]\left[\pi_1\right]\left[{}_a\Phi_{1,0}\right]\left[{}_aM_o\right] + \sum_{c=1}^{N}\left(\left[{}_a\Phi_{c,c-1}\right]\left[\pi'_c\right]\left[{}_a\Gamma_{c,c-1}\right]\left[\Lambda_c\right]\left[{}_a\Phi_{c,c-1}\right]\left[{}_aP_{c-1}\right] + \left[\pi'_c\right]\left[\Delta_{c,c-1}\right]\left[Q_{c-1}\right]\left[\Delta_{c,c-1}^T\right]\right)\right\}$$

The dimensions of the zero matrices are defined below..

Dim $[{}_1 0]$ = (3x2) Dim $[{}_3 0]$ = (2x2) Dim $[{}_5 0]$ = (1x2)

Dim $[{}_2 0]$ = (6x1) Dim $[{}_4 0]$ = (2x1)

Note: Q_k is saved at $t = t_c$ and saved until the next time through this block at which time it is Q_{c-1}.

3.4.7.2.2 Compute Covariance of Perturbation State Vector - Block VII. 2

INPUT: ${}_a\Gamma_{c,c-1}$ (9x2), Λ_c (2x9), ${}_a\Phi_{c,c-1}$ (9x9), $\underline{c}'$ (6x1), ${}_aP_{c-1}$ (9x9), $\Delta_{c,c-1}$ (9x3), Q_{c-1} (3x3)

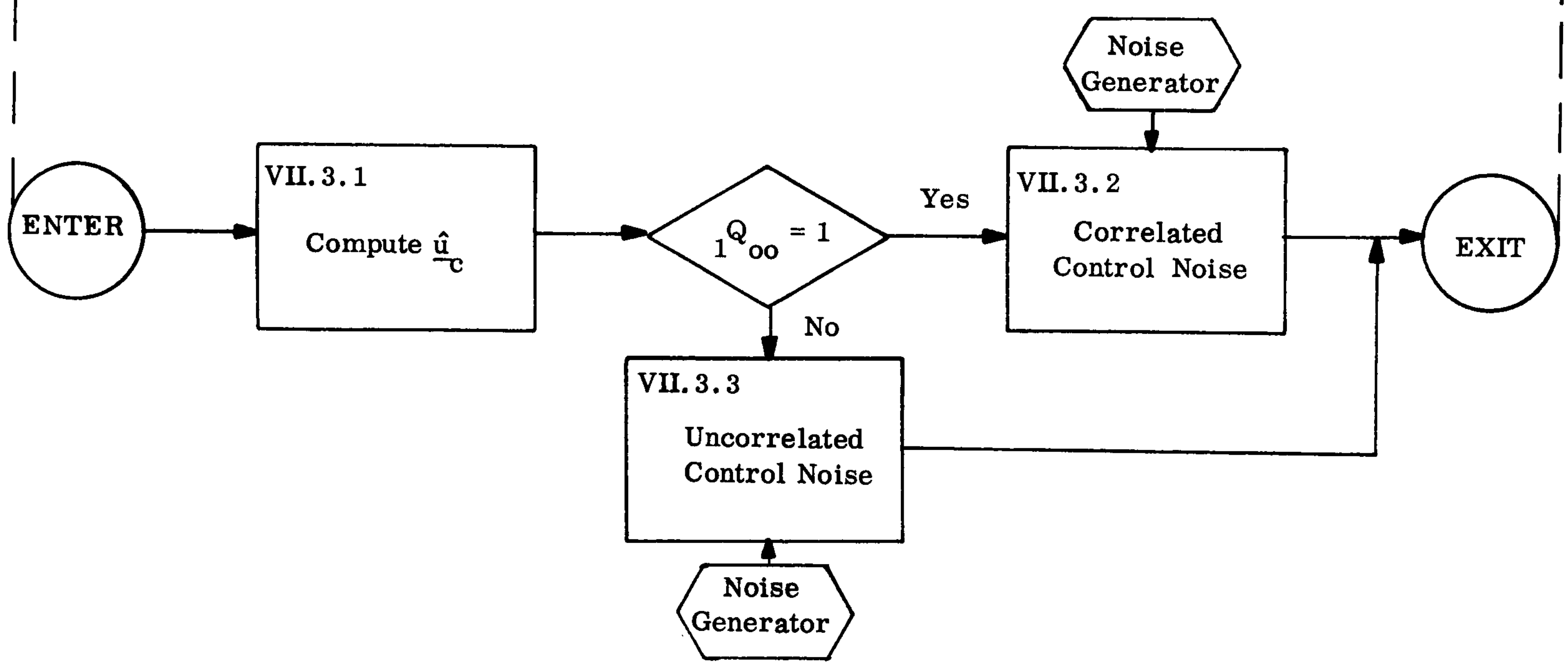
OUTPUT: ${}_aM_c$ (9x9)

$$
{}_aM_c = (I - {}_a\Gamma_{c,c-1}\Lambda_c)\,{}_a\Phi_{c,c-1}({}_aM_{c-1} - {}_aP_{c-1})\,{}_a\Phi^T_{c,c-1}\,(I - {}_a\Gamma_{c,c-1}\Lambda_c)^T
$$

$$
+ ({}_a\Phi_{c,c-1})({}_aP_{c-1})({}_a\Phi^T_{c,c-1}) \quad + \Delta_{c,c-1}Q_{c-1}\Delta^T_{c,c-1}
$$

Stored	Computed
${}_1Q_c$ (2x2)	Λ_{c+1} (2x9)
$\underline{c}$ (6x1)	${}_a\underline{x}_c$ (9x1)
${}_1Q_{oo}$	${}_a\Phi_{c+1,\,c}$ (9x9)
	$\Phi_{o,\,N}$ (6x6)

Output
$\underline{u}_c$ (2x1)



3.4.7.2.3 Compute Control Vector $\underline{u}_c$ - Block VII.3

Block VII. 3. 1 Compute Control Vector $\hat{\underline{u}}_c$

INPUT: $\underline{c}'$ (6x1), Λ_{c+1} (2x9), $_a\hat{\underline{x}}_c$ (9x1), $_a\Phi_{c+1,c}$ (9x9), $\Phi_{o,N}$ (6x6)

OUTPUT: $\underline{u}_c$ (2x1)

Note: $\Phi_{c+1,c}$ (6x6) is the upper left submatrix of $_a\Phi_{c+1,c}$ (9x9).

1) $\Phi_{c+1,o} = \Phi_{c+1,c}\,\Phi_{c,o}$

2) $\Phi_{c+1,N} = \Phi_{c+1,o}\,\Phi_{o,N}$

3) $\underline{c}'' = \Phi_{c+1,N}\,\underline{c}'$

4) $\underline{c}''' = \begin{bmatrix} \underline{c}'' \\ \\ {}_1 0 \end{bmatrix}$; Dim $[{}_1 0]$ is (3x1)

5) $\hat{\underline{u}}_c = -\Lambda_{c+1}\left\{\left[{}_a\Phi_{c+1,c}\right]\left[{}_a\hat{\underline{x}}_c\right] - \underline{c}'''\right\}$

Block VII. 3. 2 Correlated Control Noise

INPUT: ${}_1Q_c$ (2x2) , $\underline{\hat{u}}_c$ (2x1)

OUTPUT: $\underline{u}_c$ (2x1)

1) ${}_1Q_c$ is sent to the Triangularization Subroutine. Output of this routine is a diagonal matrix D_Q (2x2) and a lower triangular matrix T_Q (2x2).

2) Using the noise generator and the diagonal elements of D_Q as variances of the elements, generate two gaussian random numbers with zero means ${}_iw'_c$ $i = 1, 2$

3) Compute

$$\underline{w}_c = T_Q \, \underline{w}'_c$$

4) $\underline{u}_c = \underline{\hat{u}}_c + \underline{w}_c$

Block VII. 3. 3 Uncorrelated Control Noise

INPUT: ${}_1Q_c$ (2x2), $\underline{\hat{u}}_c$

OUTPUT: $\underline{u}_c$

1) Using the noise generator and the diagonal elements of ${}_1Q_c$ as variances of the elements, generate two gaussian random numbers with zero means ${}_iw_c$ i = 1, 2

2) $\underline{u}_c = \underline{\hat{u}}_c + \underline{w}_c$

4.0 USER'S GUIDE

4.1 INTRODUCTION

The purpose of this section is to provide whatever information is required to operate this program to its full capacity. It is intended that this information be supplied in as easy to use a form as possible. With this in mind this section was organized with a general description of the tapes, matrix input format, time point definition, and units preceding paragraphs, containing specific input instructions. The information contained in the general description is applicable to all the input. The specific input instructions relate to all the input which can be made to the program. The approach is taken that, in order to make a computer run, certain input must be supplied to the form of intermediate tapes (see 4.1.1) or data supplied on cards. The required input is established by following the specific instructions which are listed in the same order as the input on the load sheets. The load sheets are included near the description.

The various ways of operating the program are described in Paragraph 4.2.2. It is suggested that the user turn to this area, establish the operational mode designation for the type of run that is desired, and use the description which is presented in order to determine what input is required, both tape and card.

4.1.1 Description of Tapes

Tapes are referred to in this program by two names: These are intermediate tapes, which are tapes containing data used in the performance assessment part of the program; and output tapes, which contain the performance assessment output of the program.

4.1.1.1 Intermediate Tapes

There are three intermediate tapes and they all have the same general format. The first record consists of alpha numeric characters which identify the input used in generating this tape as well as quantities which were evaluated in the initialization of the program. It is a "key" for the data just described which then appears in the second record. The third record is another key which is used to identify the data which is tabulated on all succeeding records. Records 1 and 3 are included for the convenience of the user in identifying the data stored on these tapes when or if the tapes are edited and the data printed out. The intermediate tapes consist of:

1. Tape 1 - Nominal and Linear System Matrices.

   This tape contains the nominal trajectory and linear system matrices data which is required later in the program. The linear system matrices are required in the performance assessment part of the program, but the nominal trajectory data is needed to generate the other intermediate tapes as well as being required in the performance assessment. Nominal trajectory data is stored at t_G time points while the linear system matrices are stored at t_p time points. (See Section 4.3 for time point information.)

2. Tape 1' - IMU Error Matrices.

   This tape contains all of the information tabulated on tape 1 in addition to IMU error matrices which are added to the records at time $t = t_p$. For one nominal trajectory there may be several 1' tapes where the difference between the tapes is just IMU data. Since the IMU data is added after the data at a t_p time point, the locations of the nominal trajectory and linear system matrices is unchanged in a tape 1 and a tape 1' at that time point. As a consequence, tape 1' or tape 1 may be used interchangeably in the various modes of operation as long as the mode does not require the use of IMU error matrices such as performance assessment mode using the IMU with instrument errors as one of the sensors.

3. Tape 3 - Guidance Law Matrices.

   This tape contains the guidance matrices which are tabulated at t_c time points where the t_c are subsets of the t_k time points. Again, for one nominal trajectory there may be several guidance tapes. This tape is unique in that data is stored starting at the end time point proceeding to time $t = t_o$. This arrangement is due to the fact that data is stored at the time it is generated in order to get away from storage problems, and the guidance matrices must be calculated starting with terminal conditions and finishing with the initial values.

4.1.1.2 Output Tape

The format and function of the output tape is different than that of the intermediate tapes. An output tape is generated during all operational modes of the program, but the data stored during the time that intermediate tapes are being generated, called supplementary data, is not required to evaluate a performance assessment run. It is data in excess of the data stored on the intermediate tapes and is used to diagnose program ills or explain unusual output. If no intermediate tapes are generated, i.e., these tapes are input, only performance assessment data is stored on the output tape.

An example of the data arrangement on an output tape is presented below for a series of runs in which the first run of the series is a mode 8 and the remaining runs are mode 15. (See Paragraph 4.2.2.)

Run 1			
A. Intermediate Tape Input	B. Supplementary Data	C. Total Input Required for this Performance Assessment Including A	D. Performance Assessment

Run 2		Last Run	
E. Total Input Required for this Performance Assessment Including A.	F. Performance Assessment	G. Total Input Required for this Performance Assessment Including A.	H. Performance Assessment

A list of both the supplementary and the performance assessment data is given in 4. 9 and consists of that data with the highest rank number. The intermediateinput is put in A so that when the supplementary data which follows is printed, the data may be printed before the print of the supplementary data without rewinding the output tape. This is also the reason why C, E, and G contain the total card input that was required to make that particular performance assessment run. The total card input includes that data which was used to generate the intermediate tapes.

4. 1. 2 Matrix Input Format

There are three types of matrices which are supplied as input to this program. They are:

1. Nonsymmetric
2. Symmetric, constant
3. Symmetric, time varying

There is only one nonsymmetric matrix, M_{IMU}, which is input and all 9 elements must be supplied. There are a number of the symmetric constant or time invariant matrices. These may or may not be diagonal matrices. A diagonality flag for each of these matrices is supplied. If the flag equals 0, only the diagonal elements are picked up and the matrix is set up as a diagonal matrix. If the flag = 1, the matrix is nondiagonal and the program picks up the diagonal elements first, followed by the off-diagonal elements which appear in the upper triangle. The lower off-diagonal elements are set up with the program using the property of symmetry.

The third class of matrices have the same properties as the second class and are treated and input the same way with one exception. The exception is that time is the first item in each array and there may be more than one matrix, with different times, input to the program. The matrices are input with time monotonically increasing and a matrix is used until the current time in the program is greater than the input value. Assume that there is input room for ten matrices with different time arguments and only the first four have been input. If the current time exceeds the time input on the fourth matrix, for example, that matrix will be used for the remainder of the run.

4.1.3 Time Point Definition

There are six different types of time points defined throughout this program. These are the nominal control, t_G; minimum observation, t_P; actual observation, t_k; guidance, t_c; store on output tape, t_p; and the output tape edit, t_w time points. All of these sets of time points are defined by means of the following input: T_{xi} and Δt_{xi} (i=1, 2, . . . , 10).

> t_{xi} are defined in the interval $T_{xi-1} \leq t < T_{xi}$ by starting at T_{xi} and proceeding backward, in equal intervals of Δt_{xi} to the first point in the interval where $t > t_{xi-1}$. T_{xi} must be input as an integer multiple of Δt_{xi}.

The tables of T_{xi} and Δt_{xi} need not be filled but should be monotonically increasing at the last value should be greater than the end time of the program. If, and only if, the table is filled and time exceeds the last table value, the program will continue to use the last set of values to compute the t_x time points.

The t_G time points are times at which nominal control was generated in the nominal trajectory program. Nominal trajectory data is stored on tape 1 at these times for use in the actual trajectory. The t_P times are time points at which linear system matrices are stored on tape 1. This must also be a t_G time point and both data sets are stored at $t = t_P$.

The t_k time points, times when observations by the sensors are called for, are a subset of the t_p time points since linear system matrices are required in the navigation which also occurs at $t = t_k$. Finally, the control times, t_c, are a subset of the t_k time points since control is prefaced by an observation and navigation. The only restriction on the t_P time points is that they occur at t_G time points which implies that data may be written on the output tape, at $t = t_p$, between observations or control.

4.1.4 Units

The only units which are invariant in this program are those which measure angles. These must be radians. For all other calculations, any set of mutually consistent units may be used. There is no transformation between the input, computational or output quantities so the output is expressed in the same set of units as the input.

4.2 MAIN CONTROL

This section of the input identifies the intermediate and the output tapes and is used to specify the operational mode of the program. During the first run of a sequence, i.e., stacked runs, any operational mode may be called for. During subsequent runs in this sequence, only performance assessment runs may be made. It is the responsibility of the user to make whatever changes to the flags in this section which are necessary to accomplish this. If the first run was a performance assessment run, no input change is required, but if an intermediate tape or tapes were generated, the appropriate flags must be changed.

4.2.1 Tape Identification

The first four inputs to this section, 9 1 to 9 4 are used to identify either the run on a tape or the tape itself. In the first case, RUN NO. is used to distinguish one run on an output tape from other runs on the same tape. This is accomplished by taking the input, RUN NO., and storing with the input associated with that run. This identification is compared with an input to the tape edit routine, RUN NO., which specified which runs on the output tape are to be printed or edited. The three remaining inputs 9 2 to 9 4 are precautionary in nature and operate under the following rules.

a. New Tape 1 NO.

   If a new nominal trajectory and linear system matrices tape with or without IMU error matrices, i.e., tape 1 or 1' respectively, is generated during the run associated with this input, the number in 9 2 is the identification of that tape.

b. Old Tape 1 NO.

   If a tape 1 or 1' is mounted on the tape units to be used for making performance assessments or generating other intermediate tapes, the identification of the mounted tape is compared with this input quantity. If the two do not agree, an appropriate error message is printed and the run terminated. (See Paragraph 4.11.)

c. Tape 3 NO.

   If a new guidance law tape is being generated this input identifies the tape. If a previously obtained guidance law tape is mounted for use in a performance assessment, its identification must be the same as this input number.

The purpose of the last three inputs is to preclude the possibility of the wrong tape being used during the operation of this program. If this danger is not significant or if the bookkeeping is too laborious, one number, i.e., 0, may be used for all tapes in which case the tests will always be passed.

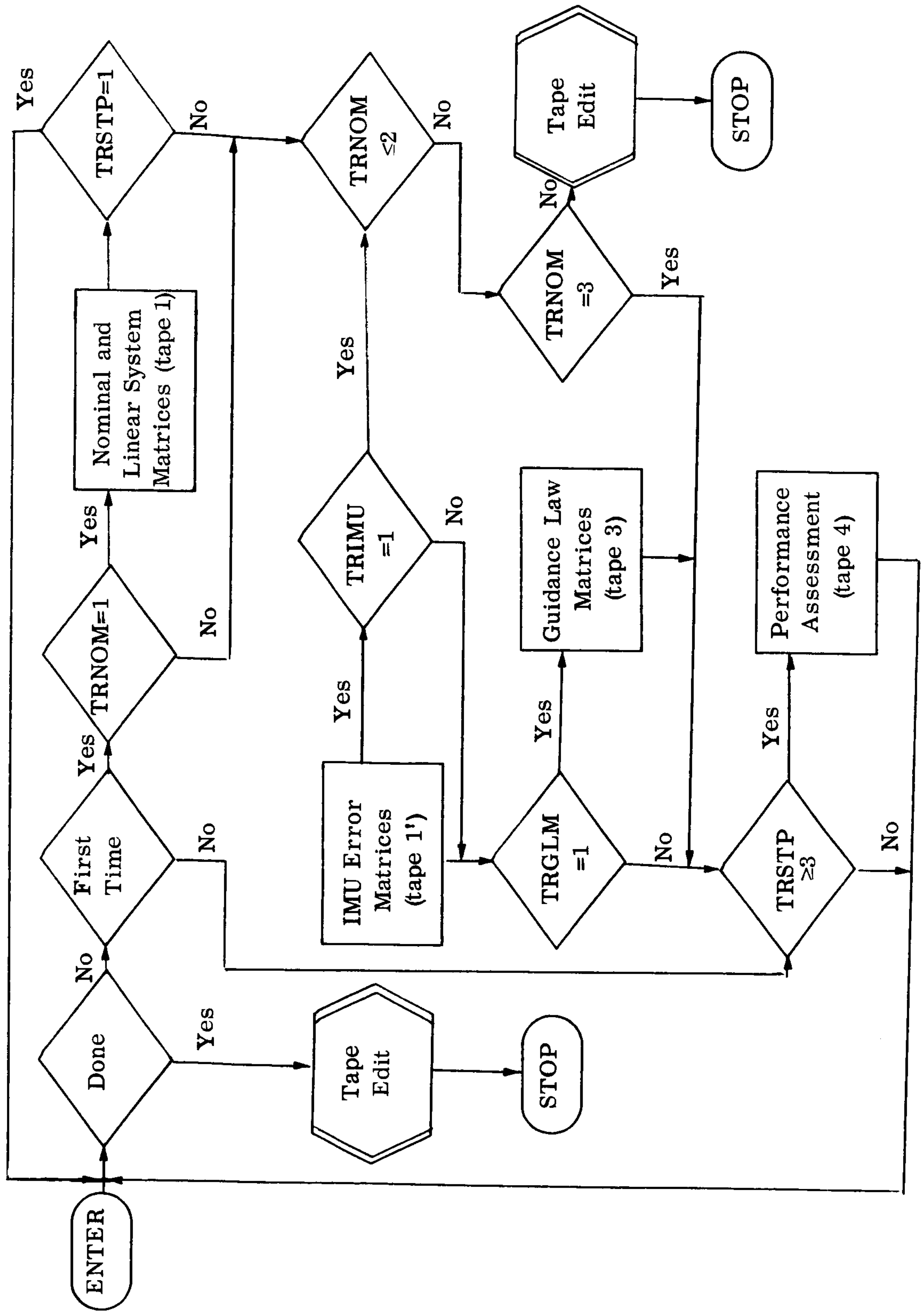


Figure 1 . Schematic of the Operational Modes of the Re-entry Performance Assessment Program

Mode	TRNOM	TRIMU	TRGLM	TRSTP	Tapes Generated	Input Tapes Req'd	Card Input Req'd	Contents of Output Tape (see notes for explanation)
1	1	0	0	1	1, 4	None	9, 1	Sup (NOM)
2	1	1	0	2	1', 4	None	9,1,2	Sup (NOM), Sup (IMU)
3	1	0	1	2	1, 3, 4	None	9,1,3	Sup (NOM), Sup (GLM)
4	1	1	1	2	1', 3, 4	None	9,1,2,3	Sup (NOM), Sup (IMU), Sup (GLM)
5	1	0	0	3	1, 4	3*	9,1,4,5,6	PA (NAV ONLY), Sup (NOM)
6	1	1	0	3	1', 4	3*	9, 1, 2 4, 5, 6	PA (NAV ONLY), Sup NOM, Sup (IMU)
7	1	0	1	3	1, 3, 4	None	9, 1, 3, 4,5,6,7	PA, Sup (NOM), Sup (GLM)
8	1	1	1	3	1',3, 4	None	9, 1,2,3 4,5,6,7	PA, Sup (NOM), Sup (IMU), Sup (GLM)
9	2	1	0	2	1', 4	1 or 1'	9, 2	Sup (IMU)
10	2	0	1	2	3, 4	1 or 1'	9, 3	Sup (GLM)
11	2	1	1	2	1', 3,4	1 or 1'	9,2,3	Sup (IMU), Sup (GLM)

Table 1 (page 1 of 2). Operational Modes of the Re-entry Performance Assessment Program

Mode	TRNOM	TRIMU	TRGLM	TRSTP	Tapes Generated	Input Tapes Req'd	Card Input Req'd	Contents of Output Tape (see notes for explanation)
12	2	1	0	3	1', 4	1 or 1', 3	9, 2, 4 5, 6, 7	PA, Sup (IMU)
	2	1	0	3	1', 4	1 or 1', 3*	9, 2, 4 5, 6	PA (NAV ONLY), Sup (IMU)
13	2	0	1	3	3, 4	1 or 1'	9, 3, 4 5, 6, 7	PA, Sup (GLM)
14	2	1	1	3	1',3,4	1 or 1'	9, 2, 3 4,5,6, 7	PA, Sup (IMU), Sup (GLM)
15	3	0	0	3	4	1 or 1'	9, 4, 5 6, 7	PA
	3	0	0	3	4	1 or 1', 3*	9, 4, 5 6	PA (NAV ONLY)
16	4	0	0	3	None	4	9	Depends on mode when generated.

The following abbreviations are used in the table above:

Sup (NOM)	supplementary data from nominal and linear system matrices
Sup (GLM)	supplementary data from guidance law matrices
Sup (IMU)	supplementary data from IMU error matrices
PA	performance assessment run
PA (NAV ONLY)	performance assessment run using only navigation, no optimum guidance
Card Input (column 8)	number appearing in column 1 of the load sheets

Table 1 (page 2 of 2). Operational Modes of the Re-entry Performance Assessment Program

4.2.2 Operational Modes

Four flags, TRNOM, TRIMU, TRGLM, and TRSTP, are input to specify how much of the program the user wishes to go through. He can, by means of these flags, operate the program in the 16 different modes listed in Table 1. These modes and the reasons for them are described in general terms in Paragraph 3.2. Any of these modes may be used as the first run in a series of stacked runs, but only mode No. 15 may be used for subsequent runs in that series. Page 21 of the load sheets indicates the procedure for inputting data for stacked runs.

Table 1 and Figure 1 on the following pages also specify which tapes must be input to operate in a particular mode as well as the card input in excess of the tape edit input which is always required.

4.2.3 Headers

There are two headers or descriptive comments which can be used to identify the runs. Each of these consists of 10 BCI words or a total of 60 characters each. Both appear at the beginning of the output section of the printout and header No. 2 appears at the top of each page.

4.3 NOMINAL TRAJECTORY AND LINEAR SYSTEM MATRICES

The nominal trajectory block in this program is almost identical to, and does give identical results as, the atmospheric entry trajectory shaping program described in "Program Description for Nominal Atmospheric Entry Trajectory," dated 31 July 1966. The computer time required for the operation of the trajectory shaping program is considerably less than that required for the operation of this program because the linear system matrices are always calculated along with the nominal. Normally, nominal trajectories will have been generated on the trajectory shaping program prior to use in this program, and the input data deck from that program may be used directly with the following exceptions.

1. Only cards from locations 1 11 to 1 130 should be used.
2. A new nonzero input, h_ρ, at 1 103 must be added.
3. The program must be terminated on time, i.e., when $t = t_{END}$.

Although, as was previously mentioned, the input to this section may have been originated for a different program, a brief description of the input is provided below.

4.3.1 Program Flags

Five flags must be input to specify which options in the program are to be exercised. These consist of

1. TRINP - defines whether the initial position and velocity are supplied in spherical (TRINP = 0) or cartesian (TRINP = 1) coordinates.
2. TRPHASE - defines the phase of the mission at time $t = t_o$ when used as an input quantity. This flag has values ranging from 1 to 7 corresponding to (1) first supercircular phase, (2) first constant altitude phase, (3) skipout control phase, (4) free-fall phase, (5) second supercircular phase, (6) second constant altitude phase, and (7) subcircular phase, respectively.
3. TRSBCL - specifies the condition which defines the beginning of the subcircular phase. These conditions consist of speed reaching an input value V_{IN} (TRSBCL = 1) or having a negative radial acceleration when the roll angle equals 0 (TRSBCL = 0).
4. TROPGN - specifies if the gains K_1 and K_2 in the constant altitude phase are input constants (TROPGN = 0) or time varying (TROPGN = 1).
5. TRACC - specifies whether change to constant altitude is accomplished when radial speed is zero (TRACC = 0) or when radial acceleration is less than an input value and radial speed is greater than another input value (TRACC = 1).

4.3.2 Trajectory Data

The input to this section consists of that data which determines the shape of the nominal trajectory. The first subset of this data consi sts of the initial position and velocity of the vehicle in either spherical or cartesian coordinates, consistent with TRINP defined above. The nominal control times t_G are specified by means of the input T_{Gi} and Δt_{Gi} as described in Paragraph 4.1.3. These values should be shown carefully because they form the basic set of time points for the program and, as described in 4.1.3, all other time points must fall on these.

A reference set of body axes P_{Io}, YA_o, RO_o is defined with respect to the body axes at $t = t_o$ by means of input Euler angles α_{10}, α_{20}, α_{30}. This input allows the user to keep the Euler angles describing the attitude of the vehicle consistent between two runs. For example, if after examining a run, it is desired to rerun a part of that run, then the Euler angles α_1, α_2, α_3 of the first run at the time the second run starts may be input as α_{10}, α_{20}, α_{30} of the second run. The time history of α_1, α_2, α_3 will be the same thereafter in both runs. Otherwise the angles may be input as arbitrary values within the ranges of $\pm\pi$, $\pm\pi/2$, and $\pm\pi$ respectively.

The roll rate gain β_φ is used to compute the angular rate of the vehicle about the velocity vector. The angular rate, ω_φ, is computed from the equation

$$\omega_\varphi = K_\varphi (\varphi_c - \varphi)$$

where $(\varphi_c - \varphi)$ is the difference between the commanded and actual roll angle. The magnitude of the angular rate is not allowed to exceed β_φ rad/sec.

Lateral control or that control which keeps the vehicle within the trajectory plane, defined at $t = t_o$, is dictated by the value assigned to ϵ_S. When ϵ_S is 10^{-4} the vehicle remains relatively close to the trajectory plane. A value of 0.7 results in no lateral control.

T_c and r_c are required input only if the program is started in a constant altitude phase. They are respectively the time at the beginning of the phase and the radial distance that the vehicle attempts to maintain. The initial roll angle, φ_o, is always input and is the value of the roll angle at $t = t_o$. Constant roll angles, φ_{c3}, φ_{11}, φ_{21}, must be input if the respective constant attitude phases (supercircular or subcircular) are to be flown. If TRSBCL = 1, then V_{IN}, the speed at which phase 7 begins, be input.

If constant altitude phases 2 and 5 are to be used and TROPGN = 1, then gains K_{11} and K_{12} (phase 2) or K_{21} and K_{22} (phase 5) must be input. These are constants which are used throughout their respective phases in the equation which generates roll command, φ_c, so as to maintain constant altitude in the equation.

$$\varphi_c = \frac{\pi}{2} + \sin^{-1} (K_1 \Delta r + K_2 \Delta \dot{r}) + \frac{\pi}{2} e^{-K_3(t - T_c)}$$

where Δr = the difference between current radial distance and the radial distance at $t = T_c$, the beginning of the constant altitude phase

$\Delta \dot{r}$ = $\dot{r}$, the radial speed

K_3 = input used to generate lift downward during the first part of a constant altitude phase. If $K_3 > 10.$,

$$\frac{\pi}{2} e^{-K_3(t - T_c)}$$

is set to 0.

If TROPGN = 0, however, ζ_1 and τ_1 or ζ_2 and τ_2 are input rather than K_{11} and K_{12} or K_{21} and K_{22}. These correspond to damping coefficients and the period of oscillation for a second order system.

The angle of attack, α, is a constant through phase 1, 2, and 3, when it is equal to α', an input quantity, and constant through phases 4, 5, 6, and 7 when it is equal to α'', another input quantity.

Phase 3 may be divided into two intervals, (1) skipout control which exists from $t = t_3$ to $t = t'_3$ in which the roll control is computed from

$$\varphi_c = F_{10} + F_{11} (t - T'_c) + F_{12} (t - T'_c)^2$$

where $$T'_c \triangleq t_3$$

(2) modified skipout control which exists from $t = t'_3$ to the time when $r = r_s$. The roll equation is modified

$$\varphi_c = F_{20} + F_{21}(t - T'_c) + F_{22}(t - T'_c)^2$$

where $$T'_c = t'_3$$

If the program is started in phase 3, TRPHSE = 3, T'_c must be input. Under these conditions, if $t_o \geq t'_3$ the program starts in the modified or second part of phase 3.

If TRACC = 1, then phase change from supercircular to constant altitude (i.e., 1 or 5 to 2 or 6) occurs when either

$$\begin{cases} \ddot{r} \leq C_{apc}\, g_o \\ \dot{r} \geq C_{vps}\sqrt{g_o R} \end{cases} \qquad \text{where } C_{vpc} < 0$$

or $$\dot{r} = 0$$

whichever occurs sooner. This is done because in some cases when $\varphi_c > \pi/2$ the vehicle exceeds maximum g's because the change to constant altitude never occurs, i.e., with $\varphi_c > \pi/2$, $\dot{r}$ nevers becomes 0. The change of phase is made earlier to correct this situation and widen the re-entry corridor. The change is based on the criterion shown so as to make it unit independent. The radial acceleration is compared to a constant times the planet's surface gravitational acceleration and the radial speed is compared with a constant times surface circular speed.

4.3.3 Vehicle Data

The data in this section defines the vehicle configuration. It implicitly defines the L/D ratio and the ballistic coefficient M/C_DS, among other things. The drag and normal force coefficients are obtained from

$$C_D = C_{D_o} + C_2\alpha^2 + C_4\alpha^4$$

$$C_N = C_{N_\alpha} + C_3\alpha^3 + C_5\alpha^5$$

The mass, M, of the vehicle, the radius of the nose cone at the heat stagnation point, R_N, and the aerodynamic surface area, S, must be input in the appropriate units. The value of R_N is used only in the heat equations and will not affect the trajectory.

4.3.4 Physical Environment

Over half of the input to this section is optional regardless of the mission or phase. It is used to help evaluate the trajectory and consists of

1. E_i (i=0, . . , 4). This input is used to compute E_n which is a measure of the integrated acceleration the pilot is subjected to. This input is independent of units since it multiplies powers of acceleration always measured in earth g's.

2. The heat equation input. The equations describing vehicular heating at the stagnation point are shown below with the necessary input. They consist of the convective heat equation

$$q_c = \frac{C_H}{\sqrt{R_N}} \left(\frac{\rho}{\rho_o}\right)^n \left(\frac{V}{\sqrt{gr}}\right)^m$$

and the radiative heat equation

$$q_r = k_H R_N \left(\frac{\rho}{\rho_o}\right)^{p_H} C_e V^q$$

$$\text{If } \frac{V}{\sqrt{gr}} < 1.73: \; q_1 \rightarrow q; \; C_{e1} \rightarrow C_e$$

$$\text{If } \frac{V}{\sqrt{gr}} \geq 1.73: \; q_2 \rightarrow q; \; C_{c2} \rightarrow C_e$$

The remaining input does affect the shape of the trajectory. These items, consisting of the planet's surface gravitational attraction, g_o, the earth's gravitational attraction, the planet's atmosphere surface density, ρ_o, and decay factor, β', and the planet's radius, R, must be input. The correlation altitude, h_ρ, should be input as a nonzero number. It is the single input used only by the linear system matrices to compute the solution, $_c\Phi$, to the linear homogeneous differential equation for the perturbative density function

$$_c\dot{\Phi} = \frac{|\dot{r}|}{h_\rho} \, _c\Phi$$

4.3.5 Program Control

The t_p time points are specified by the input T_{pi} and Δt_{pi} using methods described in Section 4.1.3. These time points are times at which the linear system matrices are stored on tape 1. Navigation cannot be accomplished without these matrices and, as a consequence, these time points specify the minimum observation interval which may be used later in the performance assessment. Every t_p time point must fall on a t_G time point.

The time associated with the beginning of the program is t_o, an input quantity. The only way the nominal trajectory can be terminated (without an error message) is when $t = t_{END}$, an input quantity. t_{END} should be chosen carefully because the time schedules for control, observation, minimum observation, nominal control must be set up so that all of these types of time points have a value equal to t_{END}. That is, they must be set up so that there is a t_G, t_p, t_k, and t_c such that

$$t_G = t_p = t_k = t_c = t_{END}$$

Two integration step sizes and allowable time errors are input to the program. The first, δt_1 and ϵ_1, are used for all phases but the skipout, which uses the second set, δt_2 and ϵ_2.

The allowable time errors are the permissible differences in time between the current value of time used in the integration routine and the exit time from the routine. If ϵ is chosen too small the exit may never occur from the integration routine and the program gets caught in a loop. The minimum value of ϵ that can be chosen must be at least as large as 1. E-X where X is the number of digits to the right of the decimal point in the value for time when the run ends. For example, if the run ends when t = NNNNN. NNN, then $\epsilon \geq 0.001$.

4.4 IMU ERROR MATRICES

The IMU error matrices are matrices which are tabulated at t_p time points and consist of numbers, which, when multiplied by the instrument error source magnitudes, ϵ_i, represent errors in the measurement made by the IMU of integrals of acceleration. The calculation of the data in this section and the generation of a 1' tape is called for by means of flags in Section 4. 2 and should be done whenever the IMU is to be used as a sensor and the error model of the IMU is desired. It is possible to use the IMU as an aiding instrument without using the IMU error model in which case no tape 1' is generated. The converse is also true; a tape 1' may be used to define the nominal trajectory without using the IMU as a sensor. In either event, the use of the IMU is defined by a flag in Section 4. 7. 1.

When the IMU error matrices are to be calculated all the input in this input block must be supplied with the exception of the header which is optional. This input consists of:

a. TROMG - a flag used to specify whether a strapdown (TROMG = 1) or a gimballed (TROMG = 0) configuration of the IMU is desired.

b. M_{IMU} (3x3) - an orthonormal transformation which defines the orientation of the instrument axes with respect to the vehicle's body axes.

c. δt_{IM}, ϵ_I - the step size used by the integration routine will be the smaller of the input step size, δt_{IM}, or the interval between t_G time points which are defined in the nominal trajectory block. The maximum integration error, ϵ_I, should be determined with the same set of rules used for its counterpart in Section 4.3.5.

d. K'_i (i=1, 2, 3, 4, 5) - normalizing coefficients, are used to scale and properly dimension the error matrices. These constants must be consistent with the units of the error budget, ϵ_i (i=1, 2, . . . , 15) defined in the mathematical model. The use of these conversion constants permits the error budget to be input directly in convenient units. In general, K_i (i=1, 2, 3) is of the form A/B where A provides the correct dimension, rad/sec, to an element in the error matrix and B is the correct scaling into units of the 1σ error source. K_j (j=4, 5) is of the form C/B where C provides the dimensions length/(time squared) and B is as defined earlier.

e. Header - 10 BCI words may be used to identify the data on this tape if it is desired. This heading is written on record No. 2.

4.5 GUIDANCE LAW MATRICES

A guidance law tape, tape 3, must be used whenever a performance assessment run is desired. The input in this section is required in order to generate that tape. In the event that only navigation is desired, i.e., the control used throughout the generation of the actual trajectory is equal to the nominal values at the same time points, then a dummy guidance tape, tape 3*, must be used. The procedure for generating this dummy tape is given below.

The time points at which perturbative control vectors, $\underline{u}_c$, are generated is established by the input T_{ci} and Δt_{ci} (i=1, 2, . . . , 10) as described in Paragraph 4.1.3. It is mandatory that every t_c time point be a t_k time point and that t_{END}, the time at which the nominal trajectory terminates, is a t_c time point. Notice that the control times are established by input to this section, not in the performance assessment part of the program, and that a change in the desired control time requires the generation of a new tape.

If non-guided trajectories or performance assessment studies are to be made a dummy guidance law tape must be mounted on one of the tape drives. One dummy tape, tape 3*, may be used for many different nominal trajectories. The important feature of this tape is that the t_c time points occur at some time after the performance assessment run ends. The program tests at $t = t_k$ the time stored on the guidance law tape to see if it is a t_c time point. When using the dummy tape, the test is never satisfied except at $t = t_o$, but the control at this time is zero if both the initial estimate of the state $a\hat{\underline{x}}_o$ and the terminal offset $\underline{c}'$ is zero. One quick way of generating a tape 3* is by generating a dummy nominal and linear system matrices starting the program in phase 4 of the nominal and with circular orbit conditions, and stopping the program when $t = t_{END}$ is large. The integration step size would be on the order of 64 secs or larger for this type run. This tape would then be used to generate the tape 3* which would be saved.

The control, W_c^U, and the state, W'^X_c, weighting matrices are input in tables with time as the argument using the format described in Paragraph 4.1.2. The dimensions of the weighting matrices are presented to indicate what constitutes reasonable values for these quantities.

$$\text{Dim}\ \{W_c^X(i,j)\} = \frac{1}{L^2} \qquad \text{for } \begin{cases} i=1,2,3 \\ j=1,2,3 \end{cases}$$

$$\text{Dim}\ \{W_c^X(i,j)\} = \frac{1}{(L/T)^2} \qquad \text{for } \begin{cases} i=4,5,6 \\ j=4,5,6 \end{cases}$$

$$\text{Dim}\ \{W_c^X(i,j)\} = \frac{1}{L \cdot L/T} \qquad \text{for } \begin{cases} i=1,2,3 \\ j=4,5,6 \end{cases} \text{ and } \begin{cases} i=4,5,6 \\ j=1,2,3 \end{cases}$$

while the dimensions of the control waiting matrices are

$$\text{Dim}\ \{W_c^U(i,j)\} = \frac{1}{(\text{rad})^2} \qquad \text{for } \begin{cases} i=1,2 \\ j=1,2 \end{cases}$$

The control law minimizes the quadratic performance index

$$V_N = \Sigma\, \underline{x}^T W_c^X \underline{x} + \underline{u}^T W_c^U \underline{u}$$

where $\underline{x}$ is the 6x1 state vector

$\underline{u}$ is the 2x1 control vector

Increasing the value of the elements of the weighting matrices decreases the allowable deviation of the state or control from the nominal. Therefore, if it is desired that the actual and nominal trajectory have the same state at the end point, the state weighting matrices should be larger when time is near the end time point than at earlier times. If it is desired to use only roll commands, the state weighting elements for angle of attack should be large relative to those for the roll command.

A header of 10 BCI words is again available to identify this intermediate tape.

4.6 ACTUAL TRAJECTORY

Most of the input required to compute and integrate the differential equations of motion in the actual trajectory is stored on tape 1 and used by this block of the program. Additional input to this block falls in three categories:

1. It specifies the differences between the actual and nominal initial conditions, vehicle and physical environment.

2. It defines trajectory constraints.
3. It specifies observation and data storage times.

The actual trajectory is perturbed from the nominal by x_o, y_o, z_o, $\dot{x}_o$, $\dot{y}_o$, $\dot{z}_o$ if TRNIC = 0. In addition, the drag and normal force coefficients are perturbed by δC_{D_o} and δC_{N_α} while the atmospheric density is perturbed by $\delta\rho_o$. If TRNIC = 1, however, these quantities are computed by a noise generator using matrices M_o (6x6), ${}_1P_o$ (2x2), and ${}_2P_o$ (1x1) as the covariance matrices for uncertainties in the state, aerodynamic coefficients and atmospheric density respectively. Regardless of the value of TRNIC, M_o, ${}_1P_o$, and ${}_2P_o$ must be input, using the procedures described in Paragraph 4.1.2, because they are used as the covariance of the estimate in the navigation block.

The density perturbation, $\delta\rho_o$, is not constant as is δC_{D_o} and δC_{N_α}. It is computed at every t_p time point, other than $t = t_o$, with a noise generator which requires the variance ${}_2Q_p$, of $\delta\rho_o$. The variance is computed from the input k_o, k_1, k_2, k_3 and h_o using the equation

$$_2Q_{p-1} = |\dot{h}(t_{p-1})| \, (k_o + [k_1 + k_2 h(t_{p-1})]e^{-k_3[h(t_{p-1} - h_o)]}$$

where $\dot{h}(t_{p-1})$ is the radial speed at the previous t_p time point

and $h(t_{p-1})$ is the altitude at that time

The run terminates if any of six input trajectory constraints is violated along with a number indicating which constraint was violated. This number is related to the constraint in Paragraph 4.11. Three of the constraints, G_{max}, Δh_{max}, and ΔR_{max}, are tested every t_G time point. These constraints require the aerodynamic deceleration be less than an input value measured in earth g's, that the magnitude of the altitude difference between the nominal and actual trajectory at a given time be less than an input value and finally, that the distance between the nominal and actual trajectory at a given time be less than a specified magnitude. The remaining three constraints are tested only at the time at which the vehicle in the nominal trajectory begins phase 4, the free-fall phase. These constraints are included to terminate the trajectory prior to going through the free-fall if the subsequent re-entry is obviously unsatisfactory; i.e., outside the re-entry corridor as defined by γ_{min} and γ_{max}, or if the apocenter distance is excessive. Take note that the flight path angle comparison is made as the vehicle exits from the atmosphere, $\gamma > 0$, and that the re-entry corridor is defined at the entry point. Because the vehicle free-falls in a central force field the flight path angle at exit is equal in magnitude but opposite in sign to the value at entry. In short, both γ_{min} and γ_{max} are input as positive numbers with radian units.

The remaining input consists of data used to compute observation time points, t_k, and output tape write times, t_P. Paragraph 4.1.3 describes how this input is used. The user is reminded that the observation time points must fall on the time points to which define the minimum observation interval in the nominal trajectory and linear system matrices section. The only restriction on the tape write times, t_P, is that they fall on t_G time points which means that the output tape may have performance assessment data stored at longer or shorter intervals than defined by the observation points.

4.7 NAVIGATION SYSTEM

The input to the navigation system consists primarily of input required to define the system configuration (specify which sensors are used), describe the instruments (noise and bias on the measurements), and specify the estimate of the nine dimensional state at time $t = t_o$.

4.7.1 System Configuration Flags

These flags define the dimension of the state vector as well as specify the aiding instruments which are used at each observation time. The dimension of the state must not exceed 34 or the program will not run. The minimum dimension is obtained when the IMU is not used and no electromagnetic sensor bias errors are called for. When the bias flag, BSFG, has value of 1, bias errors on the electromagnetic observations are simulated and the dimension of the state is increased to include these errors as new state variables. When BSFG = 0 bias errors are not simulated for the electromagnetic sensors. The IMU is not an electromagnetic sensor and BSFG has no effect on the dimension of the state when the IMU is used as the only sensor. When the IMU is used (IMFG = 1) the dimension of the state is always increased by 18. If the bias errors of the IMU are not desired, a tape 1 is used instead of a tape 1', but the dimension of the state remains the same.

Four different types of aiding instruments may be specified by means of the flags in this section. If the flag for an instrument equals zero the instrument is not used in the performance assessment run. If the flag has value 1, the instrument is used except for the ground tracker flag (TRFG) which is defined in greater detail below.

1. Horizon Sensor. Three angular measurements are made consisting of two defining the local vertical and one defining half the subtended angle that the horizon of the planet generates at the position of the spacecraft. Bias errors consist of errors in these three measurements.

2. Ground Trackers. There can be as many as three of these located on the entry planet. The TRFG may have a value of 0, 1, 2, 3 corresponding to no tracker, 1 tracker, 2 trackers, or 3 trackers, respectively. There are 7 bias errors for each ground tracker. They consist of errors in the three cartesian components defining the position of the tracker and errors

in the four measurements made by the tracker; i.e., range, range rate, elevation and azimuth.

3. Radio Altimeter. Two measurements are made by this instrument. These are altitude and time rate of change of altitude. The bias errors are errors in these measurements.

4. IMU. The three measurements made by this instrument consist of integrals of aerodynamic acceleration resolved into the cartesian coordinate system. These measurements correspond to the output of three integrating accelerometers which measure nongravitational acceleration. There are 15 bias errors. These are divided into gyro drift errors and accelerometer measurement errors. The gyro drift errors are measured about each of the gyro input axes and consist of initial attitude misalignment, constant drift, and acceleration dependent drift. The accelerometer measurement errors consist of accelerometer bias errors and acceleration errors proportional to input acceleration.

4.7.2 Ground Tracker

The input in this section defines the position of the ground trackers, the noise on the measurements, and the use of the instruments whether or not an instrument called for by the appropriate system configuration flag is used based on range or time criterion.

The first two inputs, ρ^1_{max} and ρ^2_{max}, specify distances from the vehicle to the tracker beyond which the accuracy of the tracker is poor and the variances of the radial distance and angular measurements are set to 10^6.

The location of the ground trackers and the visibility criterion of each tracker are directly related to the re-entry trajectory being flown because this program is normally operated when the vehicle is close to the surface of the planet. A tracker cannot be an object below its horizon; in fact, measurements made with elevation angles less than 5 degrees are inaccurate. As a consequence, the field of view, α, of a tracker is very restricted as can be seen from figure 2 below.

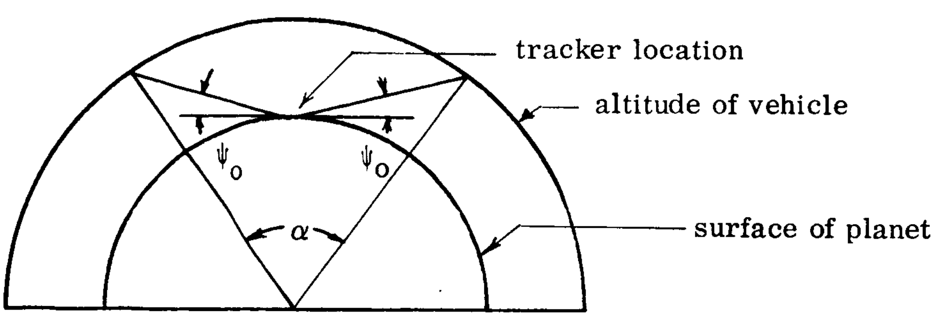


Figure 2 . Schematic Showing Relationship Between Field of View and Elevation Angle ψ_o

It is necessary to either position the trackers very carefully along the trajectory or to use a "transparent" earth if tracker measurements are desired. The transparent earth effect can be achieved by making $\psi_o = -\pi/2$. While it is true that this results in an unrealistic physical interpretation it can be used to simulate the existence of more ground trackers so that the vehicle is in the field of view of at least one tracker constantly. Another procedures which can be used to define whether a ground tracker makes an observation at a particular time during the run is that of inputting a table of ${}_iC_j$'s with time as the argument. Reference to 3.4.5.2.3 will reveal that this quantity has two functions: it is a factor of the R matrix (the covariance of noise on the measurements) and if ${}_iC_j > 10^6$, the instrument makes no observation.

The noise covariance matrices are symmetric, time-varying matrices defined by ${}_iC_j$, ${}_ia_1$, ${}_ia_2$, ${}_ia_3$, ${}_ib$, ${}_ib_o$, ${}_ib_1$, ${}_ib_2$, σ, ${}_i\psi$, $\sigma^2{}_{i\eta}$, and the cross-correlation terms. The range, $\sigma^2{}_{i\rho}$, and range rate, $\sigma^2{}_{i\dot{\rho}}$, are computed as shown below.

$$\sigma^2_{i\rho} = {}_ib_o + ({}_ib_1)({}_i\rho^2) + ({}_ib_2)({}_i\rho^4)$$

$$\sigma^2_{i\dot{\rho}} = {}_ia_o + {}_ia_1\,(1 + {}_ib\,{}_i\rho)^2\,{}_i\rho + {}_ia_2\,(1 + {}_ib_i\dot{\rho})^2\,{}_i\rho^2 + {}_ia_3\,(1 + {}_ib_i\dot{\rho})^4$$

4.7.3 Horizon Sensor

The horizon sensor is located within the space vehicle and observations are made as long as the half subtended angle is greater than the minimum permissible value, β_{min}, and less than the maximum permissible value, β_{max}. R, the radius of the planet should agree with the value input in the nominal trajectory block. The noise on the measurements is defined by a table of covariances tabulated with an argument of time. The measurements are ordered: elevation α, azimuth δ, subtended angle β^*; i.e., the R_{11} is the variance of the noise on the elevation angle.

4.7.4 Radio Altimeter

The radio altimeter is another instrument carried in the vehicle and the noise on its measurements is defined by matrix ${}_6R$ which can be input as a table with time as the argument. Altitude and radial speed are the measurements made by the instrument and they are ordered in this fashion.

4.7.5 IMU

The only input in this section consists of that required to define the magnitude of the noise on the measurements. This noise is due in part to the quantization of the output of the accelerometers.

4.7.6 Instrument Error Flags

The remainder of the input to the Navigation System section consists of data needed to generate instrument bias errors or control system noise. The instrument bias errors may be obtained in one of two ways. A noise generator may be used (TRNIB = 1) or the values may be input (TRNIB = 0). In the first case a covariance matrix must be input. In the second case the numbers themselves are input.

A diagonality flag, ${}_1Q_{OO}$, is supplied which specifies whether the covariance matrices defining noise on the control system (see 4.7.11) are diagonal or not.

Seven flags: ${}_i\sigma$ (i=1, 2, . . . , 7) are input which allow the user to generate noise on the measurements using a covariance matrix of $(1 + \sigma)R$ but use R in the Kalman filter equations which weight the observation as a function of the noise on the measurement. If the statistics of an instrument are known accurately, $\sigma = 0$ is used. On the other hand, it may be desirable to study the effect on the navigation of perfect instruments; i.e., no noise on the measurements. This can be done by making $\sigma = 1$. The same effect cannot be accomplished by making the elements of R, the covariance matrix for the instrument, equal zero because the sum of a singular matrix and R are inverted in the equation which calculates the gain matrix, K, in the Kalman filter equations. It is only when R is nonzero that the inverse can be taken. The remainder of the flags in this section are diagonality flags for the matrices defining the covariances of the bias errors.

4.7.7 Instrument Bias Covariances

Input to pertinent parts of this section must be supplied if BSFG = 1 and TRNIB = 1.

The covariance matrix for the bias errors of the i^{th} tracker is a 7x7 matrix, ${}_iB_o$, which has the following form

$${}_iB_o = \begin{bmatrix} {}_iB_I & 0 \\ 0 & {}_iB_L \end{bmatrix} \qquad i = 1, 2, 3$$

where both the ${}_iB_I$ and ${}_iB_L$ are symmetric matrices and are input using standard format. The ${}_iB_I$ are 3x3 matrices whose elements define the tracker location errors in cartesian coordinates. The ${}_iB_L$ are 4x4 matrices whose elements define the measurement errors. The measurements are ordered: range, range rate, elevation, and azimuth.

The horizon sensor bias covariance matrix ${}_4B_o$ is a 3x3 symmetric matrix whose elements are input using the standard format. The measurements are all angular and consist of elevation angle α, azimuth angle δ, and half subtended angle β^H in that order.

The radio altimeter bias covariance matrix, ${}_6B_o$, is a 2x2 symmetric matrix whose elements define the variances of bias errors in altitude and radial speed measurements respectively.

The IMU bias covariance matrix, ${}_7B_o$, is a 15x15 symmetrix matrix which can be partitioned as shown below.

$$
{}_7B_o = \begin{bmatrix} {}_7B_{G1} & 0 & 0 & 0 \\ 0 & {}_7B_{G2} & 0 & 0 \\ 0 & 0 & {}_7B_{G3} & 0 \\ 0 & 0 & 0 & {}_7B_{G4} \end{bmatrix}
$$

where ${}_7B_{Gj}$ (j=1, 2, 3) are 3x3 matrices defining the variances of the initial misalignment, constant drift, and g-dependent drift of the j^{th} gyro, and ${}_7B_{G4}$ is a 6x6 matrix defining the variances of the bias and g-dependent errors for accelerometers 1, 2, and 3.

4.7.8 Control Noise Covariance

A time-varying table of elements may be input which defines the noise on the control quantities $\delta\varphi$, roll angle. The noise is put on the commanded control rather than the actual orientation of the vehicle (in the case of the angle of attack there is no difference) and is generated at every control time, t_c. The noise remains constant between t_c time points.

4.7.9 Initial Estimate of the State

The first nine elements of the estimate of the state vector at time $t = t_o$ must be input. If no navigation mode has preceded the re-entry, the estimate is zero and no control is generated. But, if navigation has been accomplished during an interplanetary phase, the output best estimate from the interplanetary or deboost program may be input as the initial estimate resulting in guidance being generated at time $t = t_o$. If the state vector is larger than 9, all other components have an initial estimate of zero.

4.7.10 Sensor Bias Errors

If both TRNIB = 0 and BSFG = 1, then constant bias errors are input for the electromagnetic sensors called for by the system configuration flags. If both IMFG = 1 and TRNIB = 0, then constant bias errors are input for the IMU. The notation of the bias errors is defined below.

Ground Trackers

${}_i d_1, {}_i d_2, {}_i d_3$	(i=1, 2, 3)	i^{th} tracker X, Y, Z location errors
${}_i d_4, {}_i d_5, {}_i d_6, {}_i d_7$	(i=1, 2, 3)	i^{th} tracker ρ, $\dot{\rho}$, ψ, η measurement errors

Horizon Sensor

${}_4\alpha_1, {}_4\alpha_2, {}_4\alpha_3$	α, δ, β^H measurement errors

Radio Altimeter

${}_6\alpha_1, {}_6\alpha_2$	h, $\dot{r}$ measurement errors

IMU

${}_7\epsilon_1, {}_7\epsilon_2, {}_7\epsilon_3$	initial misalignment, constant drift, g-dependent drift: gyro 1
${}_7\epsilon_4, {}_7\epsilon_5, {}_7\epsilon_6$	initial misalignment, constant drift, g-dependent drift: gyro 2
${}_7\epsilon_7, {}_7\epsilon_8, {}_7\epsilon_9$	initial misalignment, constant drift, g-dependent drift: gyro 3
${}_7\epsilon_{10}, {}_7\epsilon_{11}$	bias, g-dependent errors: accel. 1
${}_7\epsilon_{12}, {}_7\epsilon_{13}$	bias, g-dependent errors: accel. 2
${}_7\epsilon_{14}, {}_7\epsilon_{15}$	bias, g-dependent errors: accel. 3

The units of the ${}_7\epsilon_i$ are dependent on the values of K'_j (j=1, 2, . . . , 5). See Section 4.4 for more details.

4.8 GUIDANCE

The only input to this section consists of an offset to the terminal point of the nominal trajectory.

4.9 TAPE EDIT INPUT

The input to this section indicates which run on the output tape, how much data from each block, and the interval at which this data is to be printed. An END card (see page 22 of the load sheets) follows the input in this section for every run which is to be printed. The run numbers specified in this section must in the same order as the run numbers on the output tape. If all the runs on the output tape have the same number and the tape edit routine inputs that run number, the tape edit routine will start printing the first run and continue printing one run for each END card in the tape edit input section.

In the event that no tape edit is desired after generating the tapes, no END card should be enclosed but there must be a FIN card. If it is desired to edit a tape that was previously generated and saved, the input to this section is standard as described previously, but TRNOM = 4 (see 4.2.2, this is mode 16).

The print code must be input as a 7-digit number. It specifies the rank number of the printout in each of the 7 blocks of the program.

The rank number specifies the desired amount of print from the output tape. Rank numbers and the corresponding print are shown below.

Block I - Nominal Trajectory Supplementary Output

Rank
- $0 \rightarrow$ No print
- $1 \rightarrow t,\ \varpi,\ \delta\varpi_c,\ \delta r,\ \dot{r},\ \ddot{r},\ r,\ \theta,\ \phi,\ V,\ \gamma,\ B,$ NEXTT3, D, N, $q_c,\ q_r,$ phase No., ϖ'_c

Block II - Linear System Matrices Supplementary Output

Rank
- $0 \rightarrow$ No print
- $1 \rightarrow A_6,\ A_6^{-1}$
- $2 \rightarrow$ Rank 1, $B'_{p,p-1},\ C'_{p,p-1},\ \Gamma'_{p,p-1},\ {}_a\mathbf{J}'_p,\ {}_2\mathbf{J}'_p,\ {}_3\mathbf{J}'_p,\ \gamma'_p$
- $3 \rightarrow$ Rank 2, $\mathbf{F}_1(t),\ \mathbf{F}_2(t),\ E_2(t),\ E_3(t),\ E_4(t),\ \dot{\Psi},\ {}_c\dot{\Phi},\ \dot{B}',\ \dot{C}',\ \dot{\Gamma}',\ \mathbf{F}_2\Phi',\ E_3\,{}_c\Phi',\ \Phi'_{p,o}$

Block III - IMU and Guidance Law Matrices

Rank
- $0 \rightarrow$ No print
- $1 \rightarrow t,\ c,\ G_{11},\ G_{21},\ G_{31},\ G_{12},\ G_{22},\ G_{32},\ W_c^U,\ W_c^X$

Block IV - Actual Trajectory Output

Rank
- 0 → No print
- 1 → t, ΔR, Δh, phase No., ${}_a\underline{x}_o$, $\underline{X}$, $\underline{X}^*$, α_1, α_2, α_3, α_1^*, α_2^*, α_3^*, ω_{PI}, ω_{YA}, ω_{RO}, ω_{PI}^*, ω_{YA}^*, ω_{RO}^*, $\underline{f}$, $\underline{f}^*$, E_n, E_n^*, q_s, q_s^*, Q, Q*, X_a, r_p, r_a
- 2 → Rank 1, r, θ, ϕ, V, γ, β, $\dot{r}$, $\ddot{r}$, D, N, q_c, q_r

Block V - Electromagnetic Sensors Output

Rank
- 0 → No print
- 1 → ${}_i\zeta$, ${}_iY$, ${}_iY^*$ (i=1, 2, 3, 4, 6, 7)
- 2 → ${}_i\underline{r}_T$, ${}_i\underline{\rho}^*$, ${}_i\dot{\underline{\rho}}^*$, ${}_i\underline{\rho}$, ${}_i\dot{\rho}$, ${}_iR$, ${}_iH_{T1}$, ${}_iH_{T2}$ (i=1, 2, 3), ${}_4R$, H^H, ${}_6R$, R^H, ${}_7R$, ${}_aJ_k$, ${}_2J_k$, ${}_3J_k$, $\int G$

Block VI - Navigation Output

Rank
- 0 → No print
- 1 → $\hat{\underline{x}}$, $\underline{x}_{Dif}$, $\tilde{\underline{x}}$, $\hat{\underline{x}}'$, ${}_i\underline{z}$ (i=1, 2, 3, 4, 6, 7)
  {[Eigenvectors, square root of Eigenvalues, volume, square root of Trace] of P_{X1}, P_{X4}, P'_{X1}, P'_{X4}}
  $\underline{b}$ ($t = t_o$ only)
- 2 → Rank 1, $\underline{x}_k$, ${}_i\underline{y}$, ${}_i\Delta\underline{y}$ (i=1, 2, 3, 4, 6, 7), P_X, P'_X, $\Phi_{k,k-1}$
- 3 → Rank 2, ${}_A\hat{\underline{x}}$, ${}_A\hat{\underline{x}}'$, ${}_c\Phi_{k,k-1}$, P_1, P_2, P_η, P_{d1}, P_{d2}, P_{d3}, $P_{\alpha 4}$, $P_{\alpha 6}$, $P_{\epsilon 7}$, P'_{d1}, P'_{d2}, P'_{d3}, $P'_{\alpha 4}$, $P'_{\alpha 6}$, $P'_{\epsilon 7}$
- 4 → Rank 3, ${}_AP$, ${}_AP'$, $B_{k,k-1}$, $C_{k,k-1}$, $\Gamma_{k,k-1}$, σ_k, ${}_c\Gamma$, Q

where the nxn error covariance matrix ${}_AP$ and the extrapolated matrix ${}_AP'$ may be partitioned as shown below.

$$
{}_A P = \begin{bmatrix}
P_X\ (6x6)\ X \longrightarrow \\
X\ P_1\ (2x2)\ X \longrightarrow \\
\longleftarrow X\ P_2\ (1x1)\ X \longrightarrow \\
\longleftarrow X\ P_\eta\ (3x3)\ X \longrightarrow \\
\longleftarrow X\ P_{d1}\ (7x7)\ X \longrightarrow \\
\longleftarrow X\ P_{d2}\ (7x7)\ X \longrightarrow \\
\longleftarrow X\ P_{d3}\ (7x7)\ X \longrightarrow \\
\longleftarrow X_{\alpha 4}\ (3x3)\ X \longrightarrow \\
\longleftarrow X\ P_{\alpha 5}\ (1x1)\ X \longrightarrow \\
\longleftarrow X\ P_{\alpha 6}\ (2x2)\ X \longrightarrow \\
\longleftarrow X\ P_\epsilon\ (15x15)\ X \longrightarrow
\end{bmatrix}
$$

and P_X can be further partitioned

$$
P_X\ (6x7) = \begin{bmatrix} P_{X1}\ (3x3) & X\ (3x3) \\ X\ (3x3) & P_{X4}\ (3x3) \end{bmatrix}
$$

Block VII - Guidance Output

$$
\text{Rank} \left\{ \begin{array}{l}
0 \rightarrow \text{No print} \\
1 \rightarrow V_N,\ \underline{u}_c,\ \hat{\underline{u}}_c,\ [\text{Eigenvectors, square root of Eigenvalues,} \\
\quad \text{volume, square root of trace] of } M_1 \text{ and } M_x \\
2 \rightarrow \text{Rank 1},\ \underline{c}'',\ {}_aM_c,\ \Lambda_c,\ \Pi_c \\
3 \rightarrow \text{Rank 2},\ \Pi',\ {}_a\Phi_{c,c-1},\ B_{c,c-1},\ C_{c,c-1},\ \Gamma_{c,c-1},\ {}_c\Gamma_{c,c-1}, \\
\quad {}_c\Phi_{c,c-1}
\end{array} \right.
$$

where the covariance of the perturbation state vector ${}_aM_c$ may be partitioned as shown

$$
{}_a\overline{M}_c = \begin{bmatrix} M_1\ (3x3) & X\ (3x3) & X\ (3x3) \\ X\ (3x3) & M_4\ (3x3) & X\ (3x3) \\ X\ (3x3) & X\ (3x3) & X\ (3x3) \end{bmatrix}
$$

The print times t_W are specified by the T_{W_i} and ΔT_{W_i} (i=1, 2, . . . , 10) as defined in Section 4. 1. 3. These time points are a subset of the $t_{\underline{p}}$ time points defined in 4. 6 since these are the times that data is stored on tape.

4.10 LOAD SHEETS

The load sheets are presented on the following pages.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 1 of 22

1 2 3 4 5	7 8 9	11 Engineer	Phone	Work Order Number	Date 72
		131,	,	,	,

MAIN CONTROL

1–5	7–9	Field 1	Field 2
9 1	DEC	RUN NO. (ident. output tape)	NEW TAPE 1 NO. (ident. new 1' tape)
9 3	DEC	OLD TAPE 1 NO. (ident. of old 1' tape)	TAPE 3 NO. (ident. of tape 3)
9 5	DEC	TRNOM (defines beginning stage)	TRIMU (compute IMU?)
9 7	DEC	TRGLM (compute GLM?)	TRSTP (defines end stage)
9 9	BCI	HEADER NO. 1 (10 BCI words)	
9 19	BCI	HEADER NO. 2 (10 BCI words)	

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 2 of 22

NOMINAL TRAJECTORY AND LINEAR SYSTEM MATRICES

PROGRAM FLAGS

1 2 3 4 5 7 8 9 11

1 11	DEC	TRINP (coord. flag) ,	TRPHSE (phase number) ,	TRSBCL (begin phase 7)
1 14	DEC	TROPGN (variable gains) ,	TRACC (test $\dot{r}$, $\ddot{r}$ flag)	

72

TRAJECTORY DATA

X_o (r_o) Y_o (λ_o) Z_o (μ_o) (initial position)

1 16	DEC	, ,

$\dot{X}_o$ (V_o) $\dot{Y}_o$ (γ_o) $\dot{Z}_o$ (A_o) (initial velocity)

1 19	DEC	, ,

T_{Gi} $i = 1, \ldots, 10$ (control interval definition time)

1 22	DEC	, , ,
1 26	DEC	, , ,
1 30	DEC	,

Δt_{Gi} $i = 1, \ldots, 10$ (nominal control calculation interval)

1 32	DEC	, , ,
1 36	DEC	, , ,
1 40	DEC	,

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 3 of 22

TRAJECTORY DATA (cont'd)

1 2 3 4 5	7 8 9	11		72
		α_{10}	α_{20}	α_{30} (initial body Euler angles)
1 42	DEC	,	,	
		K_φ (roll rate gain)	β_φ (roll rate limit)	ϵ_s (bound on out-of-plane velocity)
1 45	DEC	,	,	
		T_c (begin circular orbit)	r_c (circ. orbit rad. dist.)	φ_o (initial roll angle)
1 48	DEC	,	,	
		φ_{c3} (roll angle - ph. 7)	V_{IN} (velocity limit - ph. 7)	
1 51	DEC	,		
		K_{11} (ζ_1) (gain - ph. 2)	K_{12} (τ_1) (gain - ph. 2)	K_{13} (trans. decay - ph. 2)
1 53	DEC	,	,	
		α' (attack angle - ph. 1,2,3)	φ_{11} (roll angle - ph. 1)	
1 56	DEC	,		
		K_{21} (ζ_2) (gain - ph. 6)	K_{22} (τ_2) (gain - ph. 6)	K_{23} (trans. decay - ph. 6)
1 58	DEC	,	,	
		α'' (attack angle - ph. 4,5,6,7)	φ_{21} (roll angle - ph. 4,5)	T'_c (begin ph. 3 time)
1 61	DEC	,	,	
		F_{10}	F_{11}	F_{12} (ph. 3 control coeffs.)
1 64	DEC	,	,	
		F_{20}	F_{21}	F_{22} (modified ph. 3 control coeffs.)
1 67	DEC	,	,	

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 4 of 22

TRAJECTORY DATA (cont'd)

1 2 3 4 5	7 8 9	11		72
		r_s (begin and end ph. 4)	t_3 (begin ph. 3 time)	t'_3 (begin modified ph. 3 time)
1 70	DEC	,	,	
		C_{vpc} (test ph. 1, 5 velocity)	C_{apc} (test ph. 1, 5 acceleration)	
1 73	DEC	,		

VEHICLE DATA

		C_{DO}	C_2	C_4 (drag coeffs)
1 75	DEC	,	,	
		$C_{N\alpha}$	C_3	C_5 (normal force coeffs.)
1 78	DEC	,	,	
		M (mass of vehicle)	R_N (heat stag. pt. radius)	S (aero. area)
1 81	DEC	,	,	

PHYSICAL ENVIRONMENT

		E_i $i = 0, \ldots, 4$	(pilot g tolerance coeffs.)		
1 84	DEC	,	,	,	
1 88	DEC				
		C_H (convective heating rate constant)			
1 89	DEC				

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 5 of 22

PHYSICAL ENVIRONMENT (cont'd)

Cols 1–5	Cols 7–9	Cols 11–72
1 90	DEC	C_{E1} , C_{E2} (radiative heating rate constants)
1 92	DEC	q_1 , q_2 , p_H (radiative heating rate exponents)
1 95	DEC	k_H (radiative heating rate constant)
1 96	DEC	m , n (convective heating rate exponents)
1 98	DEC	g_o (planet's grav. accel.) , g_e (earth's grav. accel.)
1 100	DEC	ρ_o (planet's atmos. density) , β' (planet's atmos. decay factor)
1 102	DEC	R (radius of planet) , h_ρ (atmos. density var. const.)

PROGRAM CONTROL

T_{pi} i = 1, . . . , 10 (print interval definition time)

Cols 1–5	Cols 7–9	Cols 11–72
1 104	DEC	, , ,
1 108	DEC	, , ,
1 112	DEC	,

Δt_{pi} i = 1, . . . , 10 (print interval)

Cols 1–5	Cols 7–9	Cols 11–72
1 114	DEC	, , ,
1 118	DEC	, , ,
1 122	DEC	,

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 6 of 22

PROGRAM CONTROL (cont'd)

1 2 3 4 5	7 8 9	11	72
		t_o (initial time)	t_{END} (end time)
1 1 2 4	DEC	,	
		δt_1 ($\int$ step size - ph. 1,2,3,5,6,7)	ϵ_1 (max. $\int$ error - ph. 1,2,3,5,6,7)
1 1 2 7	DEC	,	
		δt_2 ($\int$ step size - ph. 4)	ϵ_2 (max. $\int$ error - ph. 4)
1 1 2 9	DEC	,	

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 7 of 22

IMU ERROR MATRICES

1 2 3 4 5	7 8 9	11			
		TROMG (strapdown or gimbal system)			
2 1	DEC				
		M_{IMU} (11)	M_{IMU} (12)	M_{IMU} (13)	(orientation
2 2	DEC	,	,		
		M_{IMU} (21)	M_{IMU} (22)	M_{IMU} (23)	matrix of
2 5	DEC	,	,		
		M_{IMU} (31)	M_{IMU} (32)	M_{IMU} (33)	instruments)
2 8	DEC	,	,		
		δt_{IM} (integ. step size)	ϵ_I (max. integ. error)		
2 1 1	DEC	,			
		K_i' (i = 1, 2, . . . , 5)	(error source normalizing coefficients)		
2 1 3	DEC	,	,		
2 1 6	DEC	,			
		HEADER (10 BCI words)			
2 1 8	BCI	,			

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 8 of 22

GUIDANCE LAW MATRICES

1 2 3 4 5	7 8 9	11	72
		T_{ci} i = 1,2,...,10 (guidance definition time)	
3 1	DEC	, , ,	
3 5	DEC	, , ,	
3 9	DEC	,	
		Δt_{ci} i = 1,2,...,10 (guidance interval)	
3 11	DEC	, , ,	
3 15	DEC	, , ,	
3 19	DEC	,	
		W_{co}^{X} (control diag. flag), W_{co}^{X} (state diag. flag)	
3 21	DEC	,	

CONTROL WEIGHTING MATRICES

		TIME	W_c^U (11)	W_c^U (22)	W_c^U (12)
* 3 24	DEC	,	,	,	
3 28	DEC	,	,	,	
3 32	DEC	,	,	,	

STATE WEIGHTING MATRICES

		TIME	W'^X_c (11)	W'^X_c (22)	W'^X_c (33)
** 3 64	DEC	,	,	,	
		W'^X_c (44)	W'^X_c (55)	W'^X_c (66)	W'^X_c (12)
3 68	DEC	,	,	,	
		W'^X_c (13)	W'^X_c (14)	W'^X_c (15)	W'^X_c (16)
3 72	DEC	,	,	,	

* Maximum number of entries is 10. This table is extended by adding 4 to the address.
** Maximum number of entries is 10. This table is extended by adding 22 to the address.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 9 of 22

GUIDANCE LAW MATRICES (cont'd)

1 2 3 4 5	7 8 9	11				72
		$W'^{X}_{c}(23)$	$W'^{X}_{c}(24)$	$W'^{X}_{c}(25)$	$W'^{X}_{c}(26)$	
3 76	DEC	,	,	,		
		$W'^{X}_{c}(34)$	$W'^{X}_{c}(35)$	$W'^{X}_{c}(36)$	$W'^{X}_{c}(45)$	
3 80	DEC	,	,	,		
		$W'^{X}_{c}(46)$	$W'^{X}_{c}(56)$			
3 84	DEC	,				

HEADER (10 BCI words)

3 285	BCI	,

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 10 of 22

ACTUAL TRAJECTORY

1 2 3 4 5	7 8 9	11			
4 1	DEC	TRNIC (noise gen. initial cond.)			
4 2	DEC	x_o	y_o	z_o	(initial state pert.)
4 5	DEC	$\dot{x}_o$	$\dot{y}_o$	$\dot{o}$	
4 8	DEC	δC_{DO}	δC_N	$\delta \rho_o$	
4 11	DEC	k_o	k_1	k_2	(constants to compute variance δ_{oo})
4 14	DEC	k_2	h_o		
4 16	DEC	G_{max} (accel. bound)	r_{ma} (max. apocenter distance)	γ_{max} (max. flight path angle)	
4 19	DEC	γ_{min} (min. flight path angle)	Δh_{max} (max. alt. difference)	ΔR_{max} (max. distance difference)	
4 30	DEC	M_{oo} (diag. flag M_o)	P_{oo} (diag. flag ${}_1P_o$)	${}_2P_o$ (atmos. density variance)	
4 33	DEC	M_o (11)	M_o(22)	M_o (33)	(variances of deviations from
4 36	DEC	M_o (44)	M_o (55)	M_o (66)	initial pos. and vel.)
4 39	DEC	M_o (12)	M_o (13)	M_o(14)	
4 42	DEC	M_o (15)	M_o (16)	M_o (23)	

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 11 of 22

ACTUAL TRAJECTORY (cont'd)

1 2 3 4 5	7 8 9	11			72
		M_o (24)	M_o (25)	M_o (26)	
4 45	DEC	,	,		
		M_o (34)	M_o (35)	M_o (36)	
4 48	DEC	,	,		
		M_o (45)	M_o (46)	M_o (56)	
4 51	DEC	,	,		
		$_1P_o$ (11)	$_1P_o$ (22)	$_1P_o$ (12)	
4 54	DEC	,	,		
		T_{ki} i = 1,2,...,10	(observation interval definition time)		
4 60	DEC	,	,	,	
4 64	DEC	,	,	,	
4 68	DEC	,			
		Δt_{ki} i = 1,2,...,10	(observation interval)		
4 70	DEC	,	,	,	
4 74	DEC	,	,	,	
4 78	DEC				
		T_{Pi} i = 1,2,...,10	(tape write definition time)		
4 80	DEC	,	,	,	
4 84	DEC	,	,	,	
4 88	DEC	,			
		Δt_{Pi} i = 1,2,...,10	(tape write interval)		
4 90	DEC	,	,	,	
4 94	DEC	,	,	,	
4 98	DEC	,			

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ____________
Page 12 of 22

NAVIGATION SYSTEM

1 2 3 4 5	7 8 9	11			
		BSFG (bias)	SSFG (S.S.)	HSFG (H.S.)	TRFG (G.T.)
5 1	DEC	,	0 ,	,	
		MINFG (ignore)	RAFG (R.A.)	IMFG (IMU)	
5 5	DEC	0 ,	,		

GROUND TRACKERS

		ρ^1_{max} (max. range for G.T. rad. dist.)		ρ^2_{max} (max. range for G.T. angular meas.)	
5 8	DEC	,			
		$_1r_T$	$_2r_T$	$_3r_T$ (tracker location radial distance)	
5 1 0	DEC	,	,		
		$_1\varphi$	$_2\varphi$	$_3\varphi$ (tracker location latitude)	
5 1 3	DEC	,	,		
		$_1\theta$	$_2\theta$	$_3\theta$ (tracker location longitude)	
5 1 6	DEC	,	,		
		$_1\psi_o$	$_2\psi_o$	$_3\psi_o$ (visibility criterion - elevation)	
5 1 9	DEC	,	,		
		TIME	$_1C_j$	$_2C_j$	$_3C_j$ (multiplies G.T. noise cov.)
* 5 2 2	DEC	,	,	,	
5 2 6	DEC	,	,	,	
5 3 0	DEC	,	,	,	
		$_1R_o$	$_2R_o$	$_3R_o$ (diagonality flag)	
5 6 2	DEC	,	,		
		$_1a_o$	$_1a_1$	$_1a_2$	$_1a_3$ (specifies var. for ρ, GT1)
5 6 5	DEC	,	,	,	
		$_2a_o$	$_2a_1$	$_2a_2$	$_2a_3$ (specifies var. for ρ, GT2)
5 6 9	DEC	,	,	,	

* Maximum number of entries is 10. This table can be expanded by adding 4 to the address.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ____________
Page 13 of 22

NAVIGATION SYSTEM (cont'd)
ELECTROMAGNETIC SENSORS (cont'd)

1 2 3 4 5	7 8 9	11 — 72	
5 73	DEC	$3^a{}_o$, $3^a{}_1$, $3^a{}_2$, $3^a{}_3$	(specifies var. for ρ, GT3)
5 77	DEC	1^b , 2b , 3b	(specifies var. for $\dot{\rho}$)
5 80	DEC	$1^b{}_o$, $1^b{}_1$, $1^b{}_2$	(specifies var. for $\dot{\rho}$, GT3)
5 83	DEC	$2^b{}_o$, $2^b{}_1$, $2^b{}_2$	(specifies var. for $\dot{\rho}$, GT2)
5 86	DEC	$3^b{}_o$, $3^b{}_1$, $3^b{}_2$	(specifies var. for $\dot{\rho}$, GT3)
5 91	DEC	$\sigma_{1\psi}{}^2$, $\sigma_{1\eta}{}^2$, $\sigma_{1\rho 1\dot{\rho}}$	(cov. for GT1)
5 94	DEC	$\sigma_{1\rho 1\psi}$, $\sigma_{1\rho 1\eta}$, $\sigma_{1\dot{\rho} 1\psi}$	
5 97	DEC	$\sigma_{1\dot{\rho} 1\eta}$, $\sigma_{1\psi 1\eta}$	
5 101	DEC	$\sigma_{2\psi}{}^2$, $\sigma_{2\eta}{}^2$, $\sigma_{2\rho 2\dot{\rho}}$	(cov. for GT2)
5 104	DEC	$\sigma_{2\rho 2\psi}$, $\sigma_{2\dot{\rho} 2\eta}$, $\sigma_{2\dot{\rho} 2\psi}$	
5 107	DEC	$\sigma_{2\dot{\rho} 2\eta}$, $\sigma_{2\psi 2\eta}$	
5 111	DEC	$\sigma_{3\psi}{}^2$, $\sigma_{3\eta}{}^2$, $\sigma_{3\rho 3\dot{\rho}}$	(cov. for GT3)
5 114	DEC	$\sigma_{3\dot{\rho} 3\psi}$, $\sigma_{3\dot{\rho} 3\eta}$, $\sigma_{3\dot{\rho} 3\psi}$	

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 14 of 22

NAVIGATION SYSTEM (cont'd)

1 2 3 4 5	7 8 9	11 $\sigma_{3\rho 3\eta}$	$\sigma_{3\psi 3\eta}$
5 117	DEC	,	

HORIZON SENSOR

		$_4R_o$ (diag. flag)	β_{min} (min sub. angle)	β_{max} (max. sub. angle)	R (radius of planet)
5 119	DEC	,	,	,	
		TIME	$_4R(11)$	$_4R(22)$	$_4R(33)$ (cov. for H.S.)
* 5 123	DEC	,	,	,	
		$_4R(12)$	$_4R(13)$	$_4R(23)$	
5 127	DEC	,	,		
		TIME	$_4R(11)$	$_4R(22)$	$_4R(33)$ (cov. for H.S.)
5 130	DEC	,	,	,	
		$_4R(12)$	$_4R(13)$	$_4R(23)$	
5 134	DEC	,	,		
		TIME	$_4R(11)$	$_4R(22)$	$_4R(33)$ (cov. for H.S.)
5 137	DEC	,	,	,	
		$_4R(12)$	$_4R(13)$	$_4R(23)$	
5 141	DEC	,	,		

RADIO ALTIMETER

		$_6R_o$ (diagonality flag for R.A.)			
5 298	DEC				
		TIME	$_6R(11)$	$_6R(22)$	$_6R(12)$ (cov. for R.A.)
** 5 299	DEC	,	,	,	
5 302	DEC	,	,	,	
5 305	DEC	,	,	,	

* Maximum number of entries is 25. This table can be expanded by adding 7 to the address.
** Maximum number of entries is 25. This table can be expanded by adding 4 to the address.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

NAVIGATION SYSTEM (cont'd)

1 2 3 4 5	7 8 9	11 … 72
		$_7R_0$ (diagonality flag for IMU)
5 399	DEC	
		TIME, $_7R_{(11)}$, $_7R_{(22)}$, $_7R_{(33)}$ (cov. for IMU)
* 5 400	DEC	, , ,
		$_7R_{(12)}$, $_7R_{(13)}$, $_7R_{(23)}$
5 404	DEC	, ,
		TIME, $_7R_{(11)}$, $_7R_{(22)}$, $_7R_{(33)}$ (cov. for IMU)
5 407	DEC	, , ,
		$_7R_{(12)}$, $_7R_{(13)}$, $_7R_{(23)}$
5 411	DEC	, ,
		TIME, $_7R_{(11)}$, $_7R_{(22)}$, $_7R_{(33)}$
5 414	DEC	, , ,
		$_7R_{(12)}$, $_7R_{(13)}$, $_7R_{(23)}$
5 418	DEC	, ,

* Maximum number of entries is 25. This table can be expanded by adding 7 to the address.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 16 of 22

NAVIGATION SYSTEM (cont'd)

1 2 3 4 5	7 8 9	11				72
		TRNIB (instrument bias errors)		$_1Q_{oo}$ (diagonality flag for $_2Q$)		
6 1	DEC	,				
		$_1\sigma$ (GT1)	$_2\sigma$ (GT2)	$_3\sigma$ (GT3)	$_4\sigma$ (H.S.)	
6 3	DEC	,	,	,		
		$_5\sigma$ (S.S.)	$_6\sigma$ (R.A.)	$_7\sigma$ (IMU)		
6 7	DEC	,	,	,		
		$_1B_{OI}$	$_2B_{OI}$	$_3B_{OI}$ (diagonality flags for $_iB_I$)		
6 10	DEC	,	,			
		$_1B_{OL}$	$_2B_{OL}$	$_3B_{OL}$ (diagonality flags for $_iB_L$)		
6 13	DEC	,	,			
		$_4B_{oo}$	$_6B_{oo}$		(diagonality flags for $_iB_o$)	
6 16	DEC	,				
		$_7B_{OG1}$	$_7B_{OG2}$	$_7B_{OG3}$	$_7B_{OG4}$ (diagonality flags for $_7B_{Gj}$)	
6 18	DEC	,	,	,		

GROUND TRACKER BIAS COVARIANCES

1 2 3 4 5	7 8 9	11				72
		$_1B_I$ (11)	$_1B_I$ (22)	$_1B_I$ (33)	(cov. tracker 1 location)	
6 22	DEC	,	,			
		$_1B_I$ (12)	$_1B_I$ (13)	$_1B_I$ (23)		
6 25	DEC	,	,			
		$_1B_L$ (11)	$_1B_L$ (22)	$_1B_L$ (33)	$_1B_L$ (44) (GT1 bias cov.)	
6 28	DEC	,	,	,		
		$_1B_L$ (12)	$_1B_L$ (13)	$_1B_L$ (14)		
6 32	DEC	,	,			
		$_1B_L$ (23)	$_1B_L$ (24)	$_1B_L$ (34)		
6 35	DEC	,	,			

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 17 of 22

NAVIGATION SYSTEM (cont'd)

Cols 1-5	Cols 7-9	Cols 11-72			
6 38	DEC	${}_2B_I$ (11) ,	${}_2B_I$ (22) ,	${}_2B_I$ (33)	(cov. tracker 2 location)
6 41	DEC	${}_2B_I$ (12) ,	${}_2B_I$ (13) ,	${}_2B_I$ (23)	
6 44	DEC	${}_2B_L$ (11) ,	${}_2B_L$ (22) ,	${}_2B_L$ (33) ,	${}_2B_L$ (44) (GT2 bias cov.)
6 48	DEC	${}_2B_L$ (12) ,	${}_2B_L$ (13) ,	${}_2B_L$ (14)	
6 51	DEC	${}_2B_L$ (23) ,	${}_2B_L$ (24) ,	${}_2B_L$ (34)	
6 54	DEC	${}_3B_I$ (11) ,	${}_3B_I$ (22) ,	${}_3B_I$ (33)	(cov. tracker 3 location)
6 57	DEC	${}_3B_I$ (12) ,	${}_3B_I$ (13) ,	${}_3B_I$ (23)	
6 60	DEC	${}_3B_L$ (11) ,	${}_3B_L$ (22) ,	${}_3B_L$ (33) ,	${}_3B_L$ (44) (GT3 bias cov.)
6 64	DEC	${}_3B_L$ (12) ,	${}_3B_L$ (13) ,	${}_3B_L$ (14)	
6 67	DEC	${}_3B_L$ (23) ,	${}_3B_L$ (24) ,	${}_3B_L$ (34)	

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 18 of 22

NAVIGATION SYSTEM (cont'd)
HORIZON SENSOR BIAS COVARIANCES

1 2 3 4 5	7 8 9	11			72
		$4^{B}o(11)$	$4^{B}o(22)$	$4^{B}o(33)$	(H.S. bias cov.)
6 70	DEC	,	,		
		$4^{B}o(12)$	$4^{B}o(13)$	$4^{B}o(23)$	
6 73	DEC	,	,		

RADIO ALTIMETER BIAS COVARIANCES

		$6^{B}o(11)$	$6^{B}o(22)$	$4^{B}o(12)$	(R.A. bias cov.)
6 77	DEC	,	,		

IMU BIAS COVARIANCES

		$7^{B}G1(11)$	$7^{B}G1(22)$	$7^{B}G1(33)$	(gyro 1 bias cov.)
6 80	DEC	,	,		
		$7^{B}G1(12)$	$7^{B}G1(13)$	$7^{B}G1(23)$	
6 83	DEC	,	,		
		$7^{B}G2(11)$	$7^{B}G2(22)$	$7^{B}G2(33)$	(gyro 2 bias cov.)
6 86	DEC	,	,		
		$7^{B}G2(12)$	$7^{B}G2(13)$	$7^{B}G2(23)$	
6 89	DEC	,	,		
		$7^{B}G3(11)$	$7^{B}G3(22)$	$7^{B}G3(33)$	(gyro 3 bias cov.)
6 92	DEC	,	,		
		$7^{B}G3(12)$	$7^{B}G3(13)$	$7^{B}G3(23)$	
6 95	DEC	,	,		

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ____
Page 19 of 22

NAVIGATION SYSTEM (cont'd)

Card (1–5)	Type (7–9)	Fields (11–72)				Comment
6 98	DEC	$7^B G4(11)$	$7^B G4(22)$	$7^B G4(33)$		(accels. bias cov.)
6 101	DEC	$7^B G4(44)$	$7^B G4(55)$	$7^B G4(66)$		
6 104	DEC	$7^B G4(12)$	$7^B G4(13)$	$7^B G4(14)$	$7^B G4(15)$	
6 108	DEC	$7^B G4(16)$	$7^B G4(23)$	$7^B G4(24)$	$7^B G4(25)$	
6 112	DEC	$7^B G4(26)$	$7^B G4(34)$	$7^B G4(35)$	$7^B G4(36)$	
6 116	DEC	$7^B G4(45)$	$7^B G4(46)$	$7^B G4(56)$		

CONTROL SYSTEM COVARIANCES

Card	Type	TIME	Q(1)	Q(22)	Q(12)	(cov. control noise)
*6 119	DEC					
6 123	DEC					
6 127	DEC					

INITIAL ESTIMATE OF STATE

Card	Type				(estimate of state $t = t_o$)
6 159	DEC	$\hat{x}_o$	$\hat{y}_o$	$\hat{z}_o$	
6 162	DEC	$\hat{\dot{x}}_o$	$\hat{\dot{y}}_o$	$\hat{\dot{z}}_o$	
6 165	DEC	$\delta\hat{C}_{Do}$	$\delta C_{N\alpha}$	$\delta\hat{C}_o$	

*Maximum number of entries is 10. The table can be extended by adding 4 to the address.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ______
Page 20 of 22

NAVIGATION SYSTEM (cont'd)
SENSOR BIAS ERRORS

1 2 3 4 5	7 8 9	11				72
6 168	DEC	$_1d_1$	$_1d_2$	$_1d_3$		(GT1 location error)
6 171	DEC	$_1d_4$	$_1d_5$	$_1d_6$	$_1d_7$	(GT1 bias errors)
6 175	DEC	$_2d_1$	$_2d_2$	$_2d_3$		(GT2 location errors)
6 178	DEC	$_2d_4$	$_2d_5$	$_2d_6$	$_2d_7$	(GT2 bias errors)
6 182	DEC	$_3d_1$	$_3d_2$	$_3d_3$		(GT3 location errors)
6 185	DEC	$_3d_4$	$_3d_5$	$_3d_6$	$_3d_7$	(GT3 bias errors)
6 189	DEC	$_4\alpha_1$	$_4\alpha_2$	$_4\alpha_3$		(HS bias errors)
6 193	DEC	$_6\alpha_1$	$_6\alpha_2$			(RA bias errors)

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 21 of 22

NAVIGATION SYSTEM (cont'd)

1 2 3 4 5	7 8 9	11				72
		$7^{\epsilon}1$	$7^{\epsilon}2$	$7^{\epsilon}3$	(Gyro 1 bias errors)	
6 195	DEC	,	,			
		$7^{\epsilon}4$	$7^{\epsilon}5$	$7^{\epsilon}6$	(Gyro 2 bias errors)	
6 198	DEC	,	,			
		$7^{\epsilon}7$	$7^{\epsilon}8$	$7^{\epsilon}9$	(Gyro 3 bias errors)	
6 201	DEC	,	,			
		$7^{\epsilon}10$	$7^{\epsilon}11$		(Accel. 1 bias errors)	
6 204	DEC	,				
		$7^{\epsilon}12$	$7^{\epsilon}13$		(Accel. 2 bias errors)	
6 206	DEC	,				
		$7^{\epsilon}14$	$7^{\epsilon}15$		(Accel. 3 bias errors)	
6 208	DEC	,				

GUIDANCE

		c'_1 (x offset)	c'_2 (y offset)	c'_3 (offset)		
7 1	DEC	,	,			
		c'_4 ($\dot{x}$ offset)	c'_5 ($\dot{y}$ offset)	c'_6 ($\dot{z}$ offset)		
	DEC	,	,			
*	END					
**	FIN					

NOTE: All numbers must have a decimal point.

*An END card must be the last input card for each run. If a series of runs is to be made, all runs are generated and the output stored on tape before the tape edit routine is used.

**A FIN card is used to specify the end of the run series. Another END and FIN card must follow these (see next page) even if no tape edit is requested.

PROGRAM 131
Single Pass/Skip Re-entry Guidance and Navigation
Performance Assessment Program

Date ________
Page 22 of 22

TAPE EDIT INPUT

1 2 3 4 5	7 8 9	11	
1 1	DEC	RUN NO. (defines run on output tape to be edited)	
1 2	DEC	PRINT CODE — Block I, Block II, Block III, Block IV, Block V, Block VI, Block VII .	(defines rank for each block)
		T_{wi} i = 1, 2, . . . , 10	(tape edit definition time)
2 1	DEC	, , ,	
2 5	DEC	, , ,	
2 9	DEC	, , ,	
		Δt_{wi} i = 1, 2, . . . , 10	(tape edit interval)
3 1	DEC	, , ,	
3 5	DEC	, , ,	
3 9	DEC	, , ,	
*	END		
**	FIN		

NOTE: All numbers must have a decimal point. (The PRINTCODE is one number)

*An END card must be the last input card for each run.

**A FIN card indicates that no more input will follow.

5.0 OPERATOR'S AND PROGRAMMER'S GUIDE: PROGRAM 131

5.1 GENERAL INFORMATION

Program 131 was written by the Los Angeles Laboratory of AC Electronics in FORTRAN IV. It was originally checked out on the IBM 7040 and then converted for use on the IBM 7094 with the machine configuration described in paragraph 5.3. It should be noted that any attempt to compile and/or execute under any system different from that described below may require modifications.

5.2 DECK ARRANGEMENT

The order of the FORTRAN decks that comprise 131.0 is shown by the compilation listing as well as by the 8 1/2 by 11 vellum 407 listing, but will also be enumerated here in condensed form along with a brief description of its function in the program and the block number of the flow chart (where applicable) in paragraph 3.3 and 3.4.

Deck	Block No.	Function
MAIN		Program control initialization
WBLK		Underflow control
FTART		Program control
FOUT1		Interface between computational routines and FQUT2
FERREX		Error monitor
FTERM		Termination control
FOUT2		Output tape writer
FNMCO	A	Input control
I93Z		Input conversions
DBLK	C	Block data ~ zeros out input matrices TW, DTW
EIGEN	C	Computes eigenvectors and eigenvalues
DIGVER	C	Prints error notes for negative eigenvalues
DEXERP	C	Prints error covariance matrix or extrapolated error covariance matrix according to rank and instrument flags
DGLAIN	C	Prints guidance law matrices input data
DHEADL	C	Prints heading at top of each page
DACTRA	C	Prints actual trajectory output
DIEDT	C	Driver for 131 edit program
DIFINT	C	Updates print time and reads in corresponding records
DIGID	C	Prints guidance output
DIGILA	C	Prints guidance law supplementary output
DIIMU	C	Prints IMU error matrices supplementary output

Deck	Block No.	Function
DLISYS	C	Prints linear system matrices supplementary output
DIMUIN	C	Prints IMU error matrices input data
DINACT	C	Prints actual trajectory input data
DINAV	C	Prints navigation data
DINEMS	C	Prints EMS input
DINGID	C	Prints guidance input
DININ	C	Reads and prints the input to the edit program and the 131 program control input
DINNAV	C	Prints navigation input
DNOTRA	C	Prints nominal trajectory supplementary output
DINPRT	C	Driver for the input subroutines
DNSTRU	C	Prints certain vectors depending on the instrument flags and the observation flags
DNUADJ	C	Adjusts input upper triangular matrices so that they may be used by INUT
DINUT	C	Functions as both IN1 and IN2 as described in Memorandum LA-3372
DREREC	C	Reads in a block of records from tape IV
DITEMS	C	Prints EMS data
DATCA	C	Finds eigenvectors, eigenvalue square roots, volumes and traces
DNOMIN	C	Prints input data for nominal trajectory and for linear systems matrices
DPAIN	C	Driver for performance assessment input subroutines
DPREIG	C	Prints eigenvectors and eigenvalue square roots
DPRTU	C	General matrix print subroutine
DRTMTL	C	Dummy matrix print subroutine that calls PRTL
DPEMES	C	Special message routine
DZETAL	C	Prints "NO OBSERVATION" message when zeta flags are zero
DTINTAL	C	Determines printout of vectors and matrices which depend on the tracker flag and on the zeta flags
FMEQ1		Matrix inversion routine
FSPINV		Interface between computational routines and SPINV
FLTMTL		Matrix multiplication routine
FMADD		Matrix addition routine
FMMOV		Matrix move routine
FTABLK		Table look-up routine

Deck	Block No.	Function
FMATHD		BCD heading construction
ARTAPE		Reads intermediate tapes
FEXPND		Forms full matrix from compressed matrix
WMMUL		Matrix multiply
FMTRN		Matrix transpose
NINTG	I.4	Integrates differential equations
MMO		Vector move
MNTM		Nominal trajectory logic
AWTAPE		Writes intermediage tapes
MNOM		Supplies running control of nominal trajectory
MNR	B.1.4	Computes reference body axes
MNC	I	Controls time dependent portion of trajectory calculation
MNG	I.1	Computes commanded roll angle
MNI	I.2	Computes dynamics of vehicle
MNH	I.2.1	Computes roll angle
MNF	I.2.2	Computes aerodynamic forces on vehicle
MNA	I.2.3	Computes acceleration of vehicle
MNT	I.3	Computes time to integrate to
MNX	I.5	Computes vehicular attitude
MNO	I.6	Makes phase change initializations
MNV	I.8	Compute evaluation equations
MNTP		No longer used
MLZ	B.2	Initializes LSM data
MLSM		Computes linear system matrices
MLF	II.1	Form state transition matrices
MLQ	II.2	Computes preliminary LSM data
MLL	II.3	Computes LSM integrands
MLM	II.4	Computes system matrices
MLT	II.5	Transform to cartesian coordinates
MND		Dummy subroutine for integration
MNZ	B.1	Initialize nominal trajectory
MNZA		Puts first BCI header on nominal tape
MNZB		Puts second BCI header on nominal tape
MNS	I.6.1A	Writes data on nominal tape
A323	III.2.3	Computation errors due to accelerometers
FSTRN	III.2.1, III.2.2	Matrices for control interval
A322	III.2.2	Acceleration errors due to gyros
A321	III.2.1	Body axes derivatives
FGDLM	III.1	Logic for guidance law matrices
ALOGIC	III.2	Logic for IMU error matrices

Deck	Block No.	Function
IZAIMU		Initializes IMU error matrices
AEVAL		Evaluation of derivatives for IMU error matrices
ICMGL	III. 1.4 III. 1.5 III. 1.6	Guidance law matrices
NPRCON		Dummy routine for NINTG
FAUGTR	III. 1.3	Augment state transition matrix
MATM		Performance assessment logic
ANGLE	V. 3. 1 V. 3. 2	Actual and nominal measurement angles
WPRODM		Computes magnitude of vector
WPROD		Computes dot product of 2 vectors
FGLOG	VII	Guidance logic
FGUID	VII. 2. 1 VII. 2. 2	Guidance computations
WORIZD	V. 3	Horizon sensor calculations
WADALT	V. 5	Radio altimeter calculations
WRAKIN	V. 1	Basic ground tracking information
WRNTRK	V. 2	Ground tracker calculations
WLECTG	V	Logic for electromagnetic sensors
F117Z		Matrix triangularization routine
LNAV	VI	Navigation control
LINNAV	B. 6	Navigation initialization
LVI2	VI. 2	Extrapolate statistics and estimate
LVI4	VI. 4	Measurement control
LVI41	VI. 4. 1	Uncorrelated noise
LVI42	VI. 4. 2	Correlated noise
LVI43	VI. 4. 3	Measurements
LVI53	VI. 5. 3	State estimation
LBVEC	B. 6. 2	Setup of instrument bias errors
LVI6	VI. 6	Compute state vector
LVI7	VI. 7	Linearity of observation matrices
WLOGIC	V	Electromagnetic sensor logic
WINITL	B. 5	Electromagnetic sensor initialization
V11Z		Random number generator
MMN		Noise generator
FTRANG		Interface between computational routines and F117Z
LMATAD		Matrix addition routine
MAG	IV. 1	Control computation - actual
MAI		Evaluation of derivatives
MAD		Dummy routine

Deck	Block No.	Function
MAH	IV. 2. 1	Computation of PHI
MAA	IV. 2. 3	Computation of acceleration
MAJ	IV. 2. 4	Atmospheric density noise
MAV	IV. 3	Compute evaluation equations
MAT	IV. 4	Actual-time control
MAX	IV. 6	Attitude computation
MAF	IV. 2. 2	Aerodynamic forces
MAZ	B. 4	Actual initialization
MACT		Performance assessment control
MAE	IV. 8	Trajectory constraints
MAS	IV. 7	Output control - actual
MAC	IV	Actual control
MAL		Intermediate tape input routine

5.3 MACHINE CONFIGURATION

Program 131 was compiled and executed on an IBM 7094 Mod II computer using the AVCO IBM 7094 IBSYS operating system supplied by NASA/ERC.

Table 5-1 contains the unit table configuration used under this system.

FUNCTION	SYMBOL	PHYSICAL	LOGICAL FORTRAN IV
Library 1	SYSLB1	A1	
Library 2	SYSLB2	Unassigned	
Library 3	SYSLB3	Unassigned	
Library 4	SYSLB4	Unassigned	
Card Reader	SYSCRD	RDA	
On-line Printer	SYSPRT	PRA	
Card Punch	SYSPCH	A0	
Output	SYSOU1	A3	6
Alternate Output	SYSOU2	A3	
Input	SYSIN1	B3	5
Alternate Input	SYSIN2	B3	
Peripheral Punch	SYSPP1	B4	7
Alt. Peripheral Punch	SYSPP2	B2	
Check Point	SYSCK1	B5	
Alternate Check Point	SYSCK2	B5	
Utility 1	SYSUT1	A4	1
Utility 2	SYSUT2	B1	2
Utility 3	SYSUT3	A2	3
Utility 4	SYSUT4	B2	4
Utility 5	SYSUT5	Unassigned	
Utility 6	SYSUT6	Unassigned	
Utility 7	SYSUT7	Unassigned	
Utility 8	SYSUT8	Unassigned	
Utility 9	SYSUT9	Unassigned	

ATTACHED UNITS NOT ASSIGNED

A5	B6
A6	B7
A7	B8
A8	B9
A9	B0

INTERSYSTEM RESERVE UNITS

None

Table 5-1. Version 13 Unit Table Configuration

5.4 PREMATURE TERMINATION OF PROGRAM

A run may be terminated prematurely due to two causes

(1) an input or a program error

(2) constraints in the actual trajectory are violated.

In either event, a special message is printed which locates the source of the problem. The message is of the form

SPECIAL MESSAGE

TIME =

RUN NUMBER =

ERROR TYPE CODE =

POSITION CODE =

SUBROUTINE CODE =

The first two items in the list above refer to the time at which the error occurs and the run number which is input in MAIN CONTROL. The remaining three items specify the type of error and may be used in conjunction with the table below to identify the program block, the subroutine deck and the sequence number of the card in that deck pertaining to the error.

Error types 2 and 3 in Table 5.2 are tape handling errors which occur when the tape read or write routines (SRTAPE and SWTAPE) detect one of three types of errors on the tape. These are

1) Incorrect run number on tape

2) End of file (or time > 1.E10) detected

3) Tape unit number out of range

"Error types" 201 to 206 inclusive are designations which specify premature program halts when these are due to trajectory constraints being violated. In addition to these messages which are printed by the tape edit routine, there are messages printed on line running from 1 to 6 which perform the same function. This data is shown in Table 5.3.

ERROR			LOCATION		
Type	Position	Subroutine	Block	Deck	Card No.
1	1	50	INPUT (A)	FNMCO	36
2	1	203	NOMINAL (I)	MNZ	454
2	2	203	NOMINAL (I)	MNZ	488
2	3	203	NOMINAL (I)	MNZ	497
2	1	215	NOMINAL (I)	MNS	130
2	2	215	NOMINAL (I)	MNS	210
2	3	215	NOMINAL (I)	MNS	234
2	4	215	NOMINAL (I)	MNS	247
2	5	215	NOMINAL (I)	MNS	260
2	1	3	GLM (III)	FGDLM	51
2	2	3	GLM (III)	FGDLM	54
2	3	3	GLM (III)	FGDLM	65
4	4	3	GLM (III)	FGDLM	86
4	5	3	GLM (III)	FGDLM	105
4	6	3	GLM (III)	FGDLM	125
2	30	3	GLM (III)	FGDLM	195
4	7	3	GLM (III)	FGDLM	206
4	8	3	GLM (III)	FGDLM	217
4	9	3	GLM (III)	FGDLM	219
2	10	3	IMU (III)	ALOGIC	106
4	11	3	IMU (III)	ALOGIC	127
2	12	3	IMU (III)	ALOGIC	130
4	13	3	IMU (III)	ALOGIC	150
2	14	3	IMU (III)	ALOGIC	164
4	15	3	IMU (III)	ALOGIC	189
2	16	3	IMU (III)	ALOGIC	192
5	17	3	IMU (III)	ALOGIC	195
2	18	3	IMU (III)	ALOGIC	198
4	19	3	IMU (III)	ALOGIC	208
6	20	3	IMU (III)	ALOGIC	221
4	21	3	IMU (III)	ALOGIC	245
2	22	3	IMU (III)	ALOGIC	248
2	23	3	IMU (III)	IZAIMU	87
100	1	111	NAVIGATION (VI)	LVI53	262
2	1	253	ACTUAL (IV)	MAZ	146
3	1	253	ACTUAL (IV)	MAZ	146
2	1	256	ACTUAL (IV)	MAC	141
3	1	256	ACTUAL (IV)	MAC	141
*201-206	2	256	ACTUAL (IV)	MAC	244
2	1	252	ACTUAL (IV)	MAL	102
3	1	252	ACTUAL (IV)	MAL	102
3	2	252	ACTUAL (IV)	MAL	135

* More complete information is presented in Table 5-3

Table 5-2 Special Messages Associated with Premature Program Stops

ERROR TYPE		TRAJECTORY CONSTRAINT
On Line	Tape Edit	
1	201	Predicted apocenter distance during free-fall is greater than an input constraint ($r_a > r_{ma}$)
2	202	Flight path angle at beginning of free-fall is greater than an input constraint ($\gamma > \gamma_{max}$)
3	203	Flight path angle at beginning of free-fall is less than input constraint ($\gamma < \gamma_{min}$)
4	204	The magnitude of the difference in altitude on the nominal and actual trajectory at a given time is greater than an input constraint ($\Delta h > \Delta h_{max}$)
5	205	The aerodynamic deceleration is greater than an input constraint ($a' > G_{max}$)
6	206	The distance between the nominal and actual trajectory at a given time is greater than an input constraint ($\Delta R > \Delta R_{max}$)

Table 5-3. Special Messages Associated with Violation of Trajectory Constraints

5.5 TAPE HANDLING

It is the purpose of this section to specify the FORTRAN tape units that are used when the program is operated in the sixteen modes of operation listed in paragraph 4.2.1. This information is presented in Table 5-4 below. As many as four program halts or pauses are programmed. The table below indicates which tapes are mounted and/or removed when these occur. The tape numbers are defined in paragraph 4.1.1, but a brief identification is also presented below.

Tape 1	-	nominal trajectory tape
Tape 1'	-	nominal trajectory and IMU error matrices tape
Tape 3	-	guidance law matrices tape
Tape 4	-	output tape

The elements of the table below consist of three symbols

a) First

1) R - remove

2) M - mount

b) Second

1) 1, 1', 3, 4, B - tape number corresponding to Tape 1, Tape 1', Tape 3, Tape 4 or a blank tape

c) Third

1) FORTRAN tape unit

For example, if mode 12 is desired, the copy tape is mounted on unit 0 at the $PAUSE. When pause 00001 occurs, tape 1' is mounted on unit 1. When pause 00002 occurs, the old or original tape 1' is removed from unit 1 and replaced with the guidance tape (tape 3). Finally, when the pause 00003 occurs, the new tape 1' is removed from unit 2, tape 3 is removed from unit 1 and the output tape (4) is removed from unit 3.

MODE	PAUSE 00001	PAUSE 00002	PAUSE 00003
1			R11, R43
2			R1'2, R43
3			R12, R31, R43
4			R1'2, R31, R43
5			R12, R31, R43
6			R1'2, R31, R43
7			R12, R31, R43
8			R1'2, R31, R43
9	M1'1		R1'1, R1'2, R43
10	M1'2		R1'2, R31, R43
11	M1'1	R1'1, MB1	R1'2, R31, R43
12	M1'1	R1'1, M31	R1'2, R31, R43
13	M1'2		R1'2, R31, R43
14	M1'1	R1'1, M31	R1'2, R31, R43
15	M1'2, M31		R1'2, R31, R43
16	M43		R43
For all modes mount copy tape on FORTRAN tape unit 0 at $PHAUSE			

Table 5-4. FORTRAN Tape Unit Utilization

6.0 REFERENCES

[1] Wingrove, R.C., "Survey of Atmospheric Re-entry Guidance and Control Methods," AIAA J. Vol. 1 No. 9, 2019-2029 (1963).

[2] "Final Report: Study of Interplanetary Transportation Systems, Phase III," Lockheed Missiles and Space Company, Contract Work Number NAS 8-2469, 2-56 to 2-109 (1964).

[3] Chapman, D.R., "An Approximate Analytical Method for Studying Entry into Planetary Atmospheres," NASA TR R-11 (1959).

[4] Chapman, D.R., "An Analysis of the Corridor and Guidance Requirements for Supercircular Entry into Planetary Atmospheres," NASA TR R-55 (1960).

[5] Chapman, D.R., Kapphahn, A.K., "Tables of Z Functions for Atmosphere Entry Analyses," NASA TR R-106 (1961).

[6] Lowe, R.E., Gervais, R.L., "Manned Entry Missions to Mars and Venus," ARS J. Vol. 32 No. 11, 1660-1668 (1962).

[7] Bryson, A.E., Denham, W.F., "Guidance Scheme for Supercircular Re-entry of a Lifting Vehicle," ARS J. Vol. 32 No. 6 894-898 (1962).

[8] Denham, W.F., "Optimal Re-entry Guidance Study Part I - Program Formulation," ASTIA No. 603240.

[9] Allen, J.H., "Aerodynamic Heating of Conical Entry Vehicles at Speeds in Excess of Earth Parabolic Speed," NASA TR R-185 (1963).

[10] Webb, P., (ed), "BioAstronautics Data Book," NASA SP-3006 (1961).

[11] Creer, B.Y., et al., "Influence of Sustained Accelerations on Certain Pilot-Performance Capabilities," Aerospace Medicine, Vol. 33, pp. 1086ff (1962).

[12] Interplanetary Guidance System Requirements Study, Volume II, Part 5, "Nominal Atmospheric Entry Trajectories - Version I", AC Electronics Division, El Segundo, California (1966).

[13] Interplanetary Guidance System Requirements Study, Volume II, Part 6, "Nominal Atmospheric Entry Trajectories - Version II", AC Electronics Division, El Segundo, California (1966).

[14] Sorenson, H. W., Salomon, Z., "Guidance and Navigation for Interplanetary Missions, Part I, Program Definition", Memorandum - LA-3027.

[15] Sorenson, H. W., "Theoretical and Computational Consideration of Linear Estimation Theory in Navigation and Guidance", Memorandum - LA-3005.

[16] Sorenson, H. W., "Controllability and Observability of Linear, Stochastic, Time-Discrete Control Systems", Memorandum LA-3464.

[17] Carlson, R. W., Swenson, B. L., "Maneuvering Flight within Earth-Entry Corridors at Hyperbolic Speeds", J. Spacecraft, Vol. 3, No. 3, March 1966.

7.0 APPENDICES

7.1 EQUATION/PROGRAM SYMBOL KEY

A list of equation symbols and their corresponding program symbols is tabulated in order to aid anyone wishing to relate the FORTRAN coding contained in Section 5.2 to the equations presented in Sections 3.3 and 3.4. This list is tabulated twice for each section; first with the equation symbols listed alphabetically and then with the program symbols alphabetically.

These keys are presented for all the blocks of the program inasmuch as one equation symbol may have been defined differently in the different blocks. The program symbol vs equation symbol list is not unique and is dependent on the block in which the symbol is used, because several programmers worked independently on the various blocks and this program uses blocks (the Electromagnetic Sensor and Navigation blocks) which were coded in part earlier.

7.1.1 Nominal Trajectory and Linear System Matrices

Equation Symbol	Program Symbol	Equation Symbol	Program Symbol
$\underline{a}$	AC, ABAR	C_N	CIV
a	FBAR	C_{N_α}	CNAL
a'	FPRI	C_{e1}	CE1
a_e	ELA	C_{e2}	CE2
A	SV (3)	C_H	CCH
A_o	RVZE (6)	C_6	CSIX
A_2	ATOO (I, J)	C_7	CSEV
A_3	ATHR (I, J)	C_8	CAIT
A_4	AFOR (I, J)	C_9	CNIN
A_5	AFIV (I, J)	C_{10}	CTEN
A_6	TRA (I, J)		
		$\underline{D}$	DEE (I) 1, X 2, Y 3, Z
B^s	OMI (I, J)		
$\dot{B}^s$	DOMI (I, J)		
$\int \dot{B}^s$	OMIP (I, J)	D	DEEM
C^s	SIG (I)	e	ELE
$\dot{C}^s$	DSIG (I)	E_i	ESUB (I), EE (I)
$\int \dot{C}^s$	SIGP (I)	E_n	ESUBN
C_{apc}	CAPC	$\dot{E}_n$	EDOTN
C_{vpc}	CVPC	E_{3c^Φ}	DXI (I)
C_D	CD	$\int E_{3c^\Phi}$	XI (I)
C_{D_o}	CDZE	E_2	EBB (I, J)
C_2	CTWO	E_3	ECBC (I)
C_3	CTHR	E_4	EDB (I, J)
C_4	CFOR		
C_5	CFIV		

Equation Symbol	Program Symbol
$\underline{f}$	AA(I)
F_0	EF0
F_1	EF1
F_2	EF2
F_{10}	EF10
F_{11}	EF11
F_{12}	EF12
F_{20}	EF20
F_{21}	EF21
F_{22}	EF22
$F_2 \Phi$	DZET(I)
$\int F_2 \Phi$	ZETA(I, J)
F_1	FA(I, J)
$IIIF_2$	FBCC(I)
IVF_2	FBD(I)
g	GEE
g_e	GEAR
g_o	GSURF
G_{max}	GMAX
h	AICH
h_ρ	HRHO
$a^J{}_p$	ORI(I, J)
$2^J{}_p$	ORJ(I, J)
$3^J{}_p$	ORK(J)

Equation Symbol	Program Symbol
k_H	SKH
K_1	CKONE
K_2	CKTWO
K_3	CKTHR
K_{11}	CK11
K_{12}	CK12
K_{13}	CK13
K_{21}	CK21
K_{22}	CK22
K_{23}	CK23
K_φ	CKPHI
M	EM
m	SMLM
n	SMLN
$\underline{N}$	EN
N	ENM
$NEXTT_i$	TNEXT(I)
p	ELP
p_H	SPH
$\underline{P}_{IO}$	PIZ(I)

Equation Symbol	Program Symbol	Equation Symbol	Program Symbol
q_1	SQ1	t_o	TZERO
q_2	SQ2	t, t_i	TNOW
q_c	SMLQC	t_3	TIME3
q_r	SMLQR	t'_3	TTPR
q_s	SMLQS	t_4	TIME4
Q	Que	t_{END}	TEND
		T_{pi}	TPR(I)
r_o	RVZE(I)	TG_i	TGO(I)
$\underline{r}$	R(I), RBAR	T_c	TIMEC
$\underline{r}_t$	RT(I)	T'_c	TCPR
r	RAD	$TRCR_i$	LFGR(I)
$\dot{r}$	RDOT		
$\ddot{r}$	RDDOT	$\underline{U}_p$	UP(I)
$\underline{r}_a$	RA(I), RVA(I)	$\underline{U}_r$	UR(I)
r_a	RADA	$\underline{U}_v$	UV(I)
r_c	RADC		
r_m	RADM	V_o	RVZE(4)
r_{ma}	RAMAX	$\underline{V}_a$	VA(I), RVA(I)
r_p	RADP	V_a	VEEA
r_s	RADS	$\underline{V}$	VBAR, V(I)
R	RSURF	$\underline{V}_t$	VT(I)
R_N	RSUBN	V	VEE
$\underline{R}_{Oo}$	ROZ(I)	V_{IN}	VIN
		$\dot{V}$	VDOT
S	ESS		

Equation Symbol	Program Symbol
X_o, Y_o, Z_o	RVZE (I) I = 1, 2, 3
$\dot{X}_o, \dot{Y}_o, \dot{Z}_o$	RVZE (I) I = 4, 5, 6
$\ddot{X}, \ddot{Y}, \ddot{Z}$	AC (I) I = 1, 2, 3
α	ALFA
α'	ALF1
α''	ALF2
α_1	AL(1)
α_2	AL(2)
α_3	AL(3)
α_{io}	ALZE (I) I = 1, 2, 3
β	BETA
β'	BATA
β_{φ}	BPHI
γ	GAMMA
γ_p	GAP(I, J)
θ	THATA
λ_o	RVZE (2)
λ	SR (2)

Equation Symbol	Program Symbol
μ	SR (3), EMU
μ_o	RVZE (3)
π_c	PIE
ρ	RHO
ρ_o	ROSUR
τ_1	CK12
τ_2	CK22
φ, φ_i	PHI
φ_{i-1}	PHIL
φ_o	PHIZ
φ_c	PHIC
φ_{c1}	PHIC1
φ_{c3}	PHIC3
φ_{11}	PHI11
φ_{21}	PHI21
ϕ	EFI
$\Phi(t_p, t_{p-1})$	CAPA (I, J)
$\Phi(t_p, t_o)$	CAPZ (I, J)
$c^{\Phi}(t_p, t_{p-1})$	CHI
$\underset{.}{c}^{\Phi^{-1}}(t, t_{p-1})$	UCHI
$c^{\Phi}(t, t_{p-1})$	DCHI
$\Phi(t_o, t_N 0$	FIZN(I, J)

Equation Symbol	Program Symbol
$\psi(t_p, t_{p-1})$	PSI(I, J)
$\dot{\psi}(t, t_{p-1})$	PSID(I, J)
ω_φ	OMPHI
ω_{PI}	OMEG(1)
ω_{RO}	OMEG(2)
ω_{YA}	OMEG(3)

Program Symbol	Equation Symbol
AA(I)	$\underline{f}$
ABAR	$\underline{a}$
AC	$\underline{a}$
AC(I) I = 1, 2, 3	$\ddot{X}, \ddot{Y}, \ddot{Z}$
AFIVE(I, J)	A_5
AFOR(I, J)	A_4
AICH	h
AL(1)	α_1
AL(2)	α_2
AL(3)	α_3
ALFA	α
ALF1	α'
ALF2	α''
ALZE(I) I = 1, 2, 3	α_{io}
ATHR(I, J)	A_3
ATOO(I, J)	A_2
BATA	β'
BETA	β
BPHI	β_φ
CAIT	C_8
CAPA(I, J)	$\Phi(t_p, t_{p-1})$
CAPC	C_{apc}
CAPZ(I, J)	$\Phi(t_p, t_o)$
CCH	C_H
CD	C_D
CDZE	C_{D_o}
CE1	C_{e1}
CE2	C_{e2}
CFIV	C_5
CFOR	C_4
CHI	$c^{\Phi}(t_p, t_{p-1})$
CK11	K_{11}
CK12	K_{12}
CK13	K_{13}
CK21	K_{21}
CK22	K_{22}
CK23	K_{23}
CNIN	C_9
CKONE	K_1
CKPHI	K_φ
CSIX	C_6
CSEV	C_7
CTEN	C_{10}
CKTHR	K_3
CKTWO	K_2
CN	C_N
CNAL	C_{N_α}
CTHR	C_3
CTWO	C_2
CVPC	C_{vpc}

Program Symbol	Equation Symbol
$\underline{D}$	DEE (I) 1, X 2, Y 3, Z
D	DEEM
DCHI	${}_c\Phi(t_p, t_{p-1})$
DELPHI	$\Delta\omega_c$
DELT1	δt_1
DELR	Δr
DELRD	$\Delta\dot{r}$
DELT2	δt_2
DGNUP(I, J)	$\dot{\Gamma}_p$
DOMI(I, J)	$\dot{B}^s$
DTGO(I)	Δt_G
DTPR(I)	Δt_p
DSIG(I)	$\dot{C}^s$
DXI(I)	$E_{3c}{}^\Phi$
DZET(I)	$F_2{}^\Phi$
EBB(I, J)	E_2
ECBC(I)	E_3
EDB(I, J)	E_4
EDOTN	$\dot{E}_n$
EE(K)	E_i
EFI	ϕ
EF0	F_0
EF1	F_1
EF2	F_2
EF10	F_{10}
EF11	F_{11}
EF12	F_{12}
EF20	F_{20}
EF21	F_{21}
EF22	F_{22}
ELA	a_e
ELE	e
ELP	p
EM	M
EMU	μ
EN	$\underline{N}$
ENM	N
EPI1	ϵ_1
EPI2	ϵ_2
EPS	ϵ_s
ESS	S
ESUB( )	E_i
FBAR	a
FBCC(I)	${}_{III}F_2$
FBD(I)	${}_{IV}F_2$
FPRI	a'
FIZN(I, J)	$\Phi(t_o, t_N)$

Program Symbol	Equation Symbol
GAMMA	γ
GAP(I, J)	γ_p
GEAR	g_e
GEE	g
GMAX	G_{max}
GNU(I, J)	Γ_p
GSURF	g_o
HRHO	h_ρ
LFGR(I)	TRCRi
OMEG(1)	ω_{PI}
OMEG(2)	ω_{RO}
OMEG(3)	ω_{YA}
OMPHI	ω_φ
OMI(I, J)	B^s
OMIP(I, J)	$\int \dot{B}^s$
ORI(I, J)	${}_aJ_p$
ORJ(I, J)	${}_2J_p$
ORK(J)	${}_3J_p$

Program Symbol	Equation Symbol
PHI	φ, φ_i
PHI11	φ_{11}
PHI21	φ_{21}
PHIC	φ_c
PHIC1	φ_{c1}
PHIC3	φ_{c3}
PHIL	φ_{i-1}
PHIZ	φ_o
PIE	π_c
PIZ(I)	$\underline{P}_{Io}$
PSI(I, J)	$\psi(t_p, t_{p-1})$
PSID(I, J)	$\dot{\psi}(t, t_{p-1})$
QUE	Q
R(I)	$\underline{r}$
RA(I)	$\underline{r}_a$
RAD	r
RADA	r_a
RADC	r_c
RADM	r_m
RADP	r_p
RADS	r_s
RBAR	$\underline{r}$
RDDOT	$\ddot{r}$
RDOT	$\dot{r}$
RHO	ρ
ROSUR	ρ_o

Program Symbol	Equation Symbol
ROZ (I)	$\underline{R}_{Oo}$
RSUBN	R_N
RSURF	R
RT (I)	$\underline{r}_t$
RVA (I)	$\underline{V}_a$
RVZE (I) I = 1, 2, 3, 4, 5, 6	X_o, Y_o, Z_o $\dot{X}_o, \dot{Y}_o, \dot{Z}_o$
SIG (I)	c^s
SIGP (I)	$\int \dot{c}^s$
SKH	k_H
SMLM	m
SMLN	n
SMLQC	q_c
SMLQR	q_r
SMLQS	q_s
SPH	p_H
SQ1	q_1
SQ2	q_2
SR (2)	λ
SR (3)	μ
SV3	A
TCPR	T'_c
TEND	t_{END}
TGO (I)	T_{Gi}
THATA	θ
TIMEC	T_c
TIME3	t_3

Program Symbol	Equation Symbol
TIME4	t_4
TNEXT (I)	$NEXTT_i$
TNOW	t, t_i
TPR (I)	T_{pi}
TRA (I, J)	A_6
TZERO	t_o
UP (I)	$\underline{U}_p$
UR (I)	$\underline{U}_r$
UU (I)	$\underline{U}_u$
UV (I)	$\underline{U}_v$
V (I)	$\underline{V}$
VA (I)	$\underline{V}_a$
VBAR	$\underline{V}$
VEDOT	$\dot{V}$
VEEA	V_a
VEND	V_{END}
VIN	V_{IN}
VT (I)	$\underline{V}_t$
XI (I)	$\int E_{ec} \Phi$
ZETA (I, J)	$\int F_2 \Phi$

7.1.2 IMU Error Matrices

Equation Symbols	Program Symbols
a_x, a_y, a_z	ACCEL
a_1, a_2, a_3	A123
C_o	CO, CX
C_o^T	CNOTT
C^T	CTMAT1
C	CMATI
C'	CP
G_{i1}	GI1
$\int G_{i1}$	GI1I
G_{i2}	GI2
$\int G_{i2}$	GI2I
K_i' $(i = 1, 2, \ldots, 5)$	AIMUK1
M	AIMUMM
M_1, M_2, M_3	MIMAT
M_1^T, M_2^T, M_3^T	MIMATT
TROMG	ATROMG
$\alpha_1, \alpha_2, \alpha_3$	ALPHA
δt_{IMU}	ADLIMU
ϵ_I	AIMUEP

Program Symbols	Equation Symbols
ACCEL	a_x, a_y, a_z
ADLIMU	δt_{IMU}
AIMUKI	K_i'
AIMUEP	ϵ_I
AIMUMM	M
ALPHA	$\alpha_1, \alpha_2, \alpha_3$
ATROMG	TROMG
A123	a_1, a_2, a_3
CMATT	C
CNOTT	C_o
CO	C_o
CP	C'
CTMATI	C^T
CX	C_o
GI1	G_{i1}
GI1I	$\int G_{i1}$
GI2	G_{i2}
GI2I	$\int G_{i2}$
MIMAT	M_i
MIMATT	M_i^T

7.1.3 Guidance Law Matrices

Equation Symbol	Program Symbol	Equation Symbol	Program Symbol
B_{c+1}, C	BC1	Π_c	PIC
$B_{c, c-1}$	BCCM1	Π_{c+1}	PIPRC
$B_{p, p-1}$	BPM1		
		$\Phi_{c, c-1}$	PHCCM1
$C_{c+1, c}$	CC1	$\Phi_{c+1, c}$	PHIC1
$C_{c, c-1}$	CCM1	$\Phi_{p, p-1}$	PHIPPM1
$C_{p, p-1}$	CPM1	$a^{\Phi}{}_{c, c-1}$	APHIC1
		$c^{\Phi}{}_{c, c-1}$	CPHCM1
t_{ci}	TM	$c^{\Phi}{}_{c+1, c}$	CPHIC1
t_{ci-1}	TMCM1		
W^U_{c-1}	WUCM1		
W^U_{co}	WUO		
W^X_c	WXC		
W^X_{co}	XWO		
$\Gamma_{c, c-1}$	GMACM1		
$\Gamma_{p, p-1}$	CMAPM1		
$a^{\Gamma}{}_{c, c-1}$	AGAMA		
$c^{\Gamma}{}_{c, c-1}$	CGMA		
$c^{\Gamma}{}_{p, p-1}$	CGMAP		
Δt_c	DLT		
Λ_c	CLAM		

Program Symbol	Equation Symbol
AGAMA	${}_a\Gamma_{c,c-1}$
APHIC1	${}_a\Phi_{c,c-1}$
BCCM1	$B_{c,c-1}$
BC1	$B_{c+1,c}$
BPM1	$B_{p,p-1}$
CCM1	$C_{c,c-1}$
CC1	$C_{c+1,c}$
CGMA	${}_c\Gamma_{c,c-1}$
CGMAP	${}_c\Gamma_{p,p-1}$
CLAM	Λ_c
CPHCM1	${}_c\Phi_{c,c-1}$
CPHIC1	${}_c\Phi_{c+1,c}$
CPM1	$C_{p,p-1}$
DLT	Δt_c
GMACM1	$\Gamma_{c,c-1}$
GMAPM1	$\Gamma_{p,p-1}$
PHCCM1	$\Phi_{c,c-1}$
PHIC1	$\Phi_{c+1,c}$
PHIPPM1	$\Phi_{p,p-1}$
PIC	Π_c
PIPRC	Π'_{c+1}

Program Symbol	Equation Symbol
TM	t_{ci}
TMCM1	t_{ci-1}
WUCM1	W^U_{c-1}
WUO	W^U_{co}
WXC	W^X_c
WXO	W^X_{co}

7.1.4 Actual Trajectory

Equation Symbol	Program Symbol	Equation Symbol	Program Symbol
a	ACC	$\int E_2$	YOTA
$\underline{a}^*$	AAN(I)	$\int E_{3\,c}{}^{\Phi}$	DXI
a'	APR	$\int E_{3\,c}{}^{\Phi}$	XI
A_2	ATOO(I, J)	$\int E_4$	UPS
A_4	AFOR(I, J)	E_i	EE(I)
A_5	AFIV(I, J)		
A_6	TRAM	$\underline{f}^*$	AA(I)
A_6^{-1}	AITA	$\int \underline{f}^*$	FEN
		$\underline{f}$	FPRN
$\dot{B}$	DOMI	$F_2{}^{\Phi}(t, t_{p-1})$	DZET
$\int \dot{B}$	OMIP	$\int F_2{}^{\Phi}$	ZETA
$\dot{C}$	DSIG	g_o	GSURF
$\int \dot{C}$	SIGP	g_e	GEAR
C_{e1}	CE1	G_{max}	GMAX
C_{e2}	CE2		
C_H	CCH	h	AICH
C_{D_o}	CDST	h^*	ACHN
C_2	CTWO	h_o	ACHZ
C_4	CFOR	h_{i-1}	ACHL
C_{N_α}	CNST	$\dot{h}_{i-1}$	DACH
C_3	CTHR	h_{max}	DELHM
C_5	CFIV		
		$a^J{}_p$	ORI
$\dot{E}_n$	EDOTN	$2^J{}_p$	ORJ
E_n	ESUBN	$3^J{}_p$	ORK
E_n^*	ENN		

Equation Symbol	Program Symbol
$k_i (i = 0, 1, 2, 3)$	SK(I)
k_H	SKH
K_φ	CKPHI
M	EM
m	SMLM
M_{oo}	SWM
M_o	AM(I)
n	SMLN
p_H	SPH
${}_1P_{oo}$	SWP
${}_1P_o$	AP, API
${}_2P_o$	P2Z
Q	QCAP
q_1	SQ1
q_2	SQ2
q_s	SMLQS
q^*_s	QSN
Q^*	QUN
R	RSURF
R_N	RSUBN
r_{ma}	RAMAX
$\underline{r}_t$	RT(I)
S	ESS

Equation Symbol	Program Symbol
TRINC	TRNIC
t_o	TZERO
t^*_i	TIM
t_i	TACT
$T_{\underline{P}i}$	TPT(I)
T_{ki}	TGA(I)
$\underline{V}_t$	VT(I)
w_ρ	WRO
$\ddot{X}, \ddot{Y}, \ddot{Z}$	AC(I)
$\dot{X}, \dot{Y}, \dot{Z}$	V(I)
X, Y, Z	R(I)
$\left.\begin{matrix} X_o, Y_o, Z_o, \\ \dot{X}_o, \dot{Y}_o, \dot{Z}_o \end{matrix}\right\}$	RVZC(I)
$\left.\begin{matrix} X^*, Y^*, Z^*, \\ \dot{X}^*, \dot{Y}^*, \dot{Z}^* \end{matrix}\right\}$	RN(I)
x_o	CXO
y_o	DYO
z_o	DZO
$\dot{x}_o$	DXDO
$\dot{y}_o$	DYDO
$\dot{z}_o$	DZDO
δC_{D_O}	DCDO
δC_{N_α}	DCNA
$\delta_{\rho o}$	DRHO

Equation Symbol	Program Symbol
α^*_i	AN(I)
α^*	ALST
β	BETA
β'	BATA
β_φ	BPHI
γ_{max}	GAMAX
γ_{min}	GAMIN
$\dot{\Gamma}$	DGNU
$\int\dot{\Gamma}$	GNUP
$_c\Gamma$	CNU
Δh	DELH
ΔR	DELP
ΔR_{max}	DELRM
Δt_{ki}	DTGA (I)
Δt_{Pi}	DTPT (I)
δt	DELTI
ϵ^*	EPN
θ	THATA
ρ_o	ROSUR

Equation Symbol	Program Symbol
φ	PHI
φ_o	PHIZ
φ^*_c	PHIN
ϕ	EFI
Φ^{-1}	SIT
$\Phi(t, t_{p-1})$	CAPA
$_c\Phi(t, t_{p-1})$	CHI
$_c\Phi^{-1}$	UCHI
$_c\dot{\Phi}$	DCHI
$\int {_c\Phi^{-1}}$	CHIP
ψ	PSI
$\dot{\psi}$	DPSI(I, J)
ω_i	OMN(I)

Program Symbol	Equation Symbol
AA(I)	$\underline{f}^*$
AAN(I)	$\underline{a}^*$
ACC	a
AC(I)	$\ddot{X}, \ddot{Y}, \ddot{Z}$
ACHL	h_{i-1}
ACHN	h^*
ACHZ	h_o
AFOR	A_4
AFIV	A_5
AICH	h
AITA	A_6^{-1}
ALST	α^*
AM(I)	M_o
AN(I)	α^*_i
AP(I)	${}_1P_o$
APR	α'
ATOO(I, J)	A2
BATA	β'
BETA	β
BPHI	β_φ
CAPA	$\Phi(t, t_{p-1})$
CCH	C_H
CDST	C_{D_o}
CE1	C_{e1}
CE2	C_{e2}
CFIV	C_5

Program Symbol	Equation Symbol
CFOR	C_4
CHI	${}_c\Phi(t, t_{p-1})$
CHIP	$\int_c \Phi^{-1}$
CKPHI	K_φ
CNST	C_{N_α}
CNU	${}_c\Gamma$
CTHR	C_3
CTWO	C_2
DACH	h_{i-1}
DCDO	δC_{D_o}
DCHI	${}_c\Phi$
DCNA	δC_{N_α}
DELH	Δh
DELHM	h_{max}
DELP	ΔR
DELRM	ΔR_{max}
DELTI	δt
DGNU	$\dot{\Gamma}$
DOMI	$\dot{B}$
DPSI(I, J)	$\dot{\psi}$
DRHO	$\delta\rho_o$
DSIG	$\dot{C}$
DTGA(I)	Δt_{ki}
DTPT(I)	Δt_{Pi}
DXI	$E_3 \, {}_c\Phi$
DXDO	$\dot{x}_o$
DXO	x_o

Program Symbol	Equation Symbol
DYDO	$\dot{y}_o$
DYO	y_o
DZDO	$\dot{z}_o$
DZO	z_o
EDOTN	$\dot{E}_n$
EE (I)	E_i
EFI	ϕ
EM	M
ENN	E^*_n
EPN	ϵ^*
ESS	S
ESUBN	E_N
FEN	$\int \underline{f}^*$
FPRN	$\underline{f}$
GAMAX	γ_{max}
GAMIN	γ_{min}
GEAR	g_e
GMAX	G_{max}
GNUP	$\int \dot{\Gamma}$
GSURF	g_o
OMIP	$\int \dot{B}$
OMN (I)	ω_i
ORI	${}_aJ_p$
ORJ	${}_2J_p$
ORK	${}_3J_p$

Program Symbol	Equation Symbol
PHI	φ
PHIN	φ^*_c
PHIZ	φ_o
PSI	ψ
P2Z	${}_2P_o$
QCAP	Q
QSN	q^*_s
QUN	Q^*
R (I)	$\underline{r}$
RAMAX	r_{max}
RN (I)	$\underline{r}^*, \underline{v}^*$
ROSUR	ρ_o
RSUBN	R_N
RSURF	R
RT (I)	$\underline{r}_t$
RVZCI	$\underline{r}_o, \underline{V}_o$
SIGP	$\int \dot{C}$
SIT	Φ^{-1}
SK(I)	k_i $(i = 0, 1, 2, 3)$
SKH	k_H
SMLM	m
SMLN	n
SMQS	q_s
SPH	p_H
SQ1	q_1
SQ2	q_2

Program Symbol	Equation Symbol
SWM	M_{oo}
SWP	${}_1P_{oo}$
TACT	t_i
TGA (I)	T_{ki}
THATA	θ
TIM	t^*_i
TPT (I)	T_{Pi}
TRAM	A_6
TRNIC	TRNIC
TZERO	t_o
UCHI	${}_c\Phi^{-1}$
UPS	$\int E_4$
V (I)	$\underline{V}$
VT (I)	$\underline{V}_t$
WRO	w_ρ
X (I)	$\int E_3 \, {}_c\Phi$
ZETA	$\int F_2 \Phi$

7.1.5 Electromagnetic Sensors

Equation Symbol	Program Symbol
${}_i a_o$	AEXP
b_i	B2EXP
${}_i b_j$	B1*XP
${}_i C_j$	CJG
${}_i H_{T1}$	HA
${}_i H_{T2}$	HB
HSFG	HSFG
IMFG	IMFG
r	RA
RAFG	RAFG
r_o	RZERO
$\left.\begin{matrix}\underline{r}_k \\ \dot{\underline{r}}_k\end{matrix}\right\}$	RAC
$\left.\begin{matrix}\underline{r}^*_k \\ \dot{\underline{r}}^*_k\end{matrix}\right\}$	RAN
TRFG	TRFG
${}_i\underline{Y}$	YA
${}_i\underline{Y}^*$	YSTA

Equation Symbol	Program Symbol
β^H_{max}	BTMAX
β^H_{min}	BTMIN
θ	THETAG
${}_o\zeta$	ZETZA
${}_i\zeta$	ZETA
${}_i\underline{\rho}_k$	RHO
${}_i\dot{\underline{\rho}}_k$	RHODOT
${}_i\underline{\rho}^*_k$	RHOSTA
${}_i\dot{\underline{\rho}}_k$	RHOSTDT
φ	PHIG
σ(matrix G. T.)	COVR

Program Symbol	Equation Symbol
AEXP	${}_i a_o$
BTMAX	B^H_{max}
BTMIN	B^H_{min}
BIEXP	b_i
BZEXP	${}_i b_j$
CJG	${}_i C_j$
COVR	Ground tracker covariance matrix
HA	${}_i H_{T1}$
HB	${}_i H_{T2}$
HSFG	HSFG
IMFG	IMFG
PHIG	φ
RA	r
RAC	$\underline{r}_k, \dot{r}_k$
RAFG	RAFG
RAN	$\underline{r}^*_k, \underline{\dot{r}}^*_k$
RHO	${}_i\underline{\rho}$
RHODOT	${}_i\underline{\dot{\rho}}$
RHOSTA	${}_i\underline{\rho}^*$
RHOSTDT	${}_i\underline{\dot{\rho}}^*$
RZERO	R
THETAG	θ
TRFG	TRFG

Program Symbol	Equation Symbol
YA	${}_i\underline{Y}$
YSTA	${}_i\underline{Y}^*$
ZETA	${}_i\zeta$
ZETZA	${}_o\zeta$

7.1.6 Navigation

Equation Symbol	Program Symbol	Equation Symbol	Program Symbol
BSFG	BSFG	$A^{P}{}_{k}, A^{P'}{}_{k}, A^{P}{}_{o}$	APK
DIMFG	DIMFG		
HSFG	HSFG	Q_k	QKAY
IMFG	IMFG	${}_1Q_k$	QTAB
RAFG	RAFG	${}_2Q_k$	Q2K
SSFG	SSFG	Q_{k-1}	QKM1
TRFG	TRFG		
TRNIB	TRNIB	${}_iR_o$	RDIAG
		${}_iR_k$	AIRK
$\underline{b}$	BVEC		
$B_{k, k-1}$	BEEKMI	${}_iT_Q$	AITQ
$\underline{c}_{k-1}$	CM1		
$C_{k, k-1}$	CEEKM1	$\hat{\underline{u}}_{k-1}$	UVECM1
		$\underline{u}_{k-1}$	UM1VEC
${}_id$	AIDEE		
${}_iD_R$	AIDK	${}_i\underline{v}_k$	AIVK
h	SMLH	w_{k-1}	WM1
h_o	SMLHO		
$\dot{h}$	HDOT	$\underline{x}_o$	XOVEC
${}_iH_k$	HONE, HTWO	$\underline{x}_{k-1}$	XM1VEC
i_kH_k	HTWO	$\underline{x}_k$	XKVEC
		$\underline{x}_{DIE}$	XDIP
k_i (i=0, 1, 2, 3)	SMLK	$\tilde{\underline{x}}_k$	XSQGK
${}_iK_k$	AIKK	$\dot{\underline{x}}_k$	AXHK
$A^{M}{}_{o}$	MA		

Equation Symbol	Program Symbol
${}_i\underline{Y}_k$	YA
${}_i\underline{Y}^*{}_k$	YSTA
${}_i\underline{z}_k$	AIZK
${}_i\underline{\alpha}$ (i=4, 5, 6)	AIALF
γ_k	GAMMAK
$\Gamma_{k, k-1}$	TAUKM1
${}_c\Gamma_{k, k-1}$	CTAKM1
${}_A\Delta_{k, k-1}$	ADELTA
${}_A\Delta^T{}_{k, k-1}$	ADELTT
${}_i\Delta\underline{y}$	DELTAY
${}_7\underline{\epsilon}$	EPSILN
${}_o\zeta$	ZETZA
${}_i\zeta$	ZETA
$\underline{\eta}_k$	ETAK
$\underline{\sigma}_k$	SIGK
${}_i\sigma$	SIGMAI
$\Phi_{k, k-1}$	PHIKM1
${}_c\Phi_{k, k-1}$	CPHKM1
${}_A\Phi_{k, k-1}$	APHIT

Program Symbol	Equation Symbol
ADELTA	$A^{\Delta}{}_{k,\,k-1}$
ADELTT	$A^{\Delta^T}{}_{k,\,k-1}$
AIALF	${}_i\underline{\alpha}$
AIDEE	${}_i\underline{d}$
AIDK	${}_iD_R$
AIKK	${}_iK_k$
AIRK	${}_iR_k$
AITQ	${}_iT_Q$
AIVK	${}_i\underline{v}_k$
APHIT	$A^{\Phi}{}_{k,\,k-1}$
APK	$A^{P}{}_k$ or $A^{P'}{}_k$
AIZK	${}_i\underline{z}_k$
AXHK	$\underline{\hat{x}}_k$
BEEKM1	$B_{k,\,k-1}$
BSFG	BSFG
BVEC	$\underline{b}$
CEEKM1	$C_{k,\,k-1}$
CM1	$\underline{c}_{k-1}$
CPHKM1	$c^{\Phi}{}_{k,\,k-1}$
CTAKM1	$c^{\Gamma}{}_{k,\,k-1}$
DELTAY	${}_i\Delta\underline{y}$
DIMFGM	DIMFG (m dimension)
DIMFGN	DIMFG(n dimension
EPSILN	${}_7\underline{\epsilon}$
ETAK	$\underline{\eta}_k$
GAMMAK	γ_k
HDOT	$\dot{h}$
HONE	${}_iH_k$
HSFG	HSFG
HTWO	${}_{i_h}H_k$
IMFG	IMFG
MA	$A^{M}{}_o$
PHIKM1	$\Phi_{k,\,k-1}$
QKAY	Q_k
QKM1	Q_{k-1}
QTAB	$1Q_k$
Q2K	$2Q_k$
RAFG	RAFG
RDIAG	${}_iR_o$
SIGK	$\underline{\sigma}_k$
SIGMAI	${}_i\sigma$
SMLH	h
SMLHO	h_o
SMLK	k_i
SSFG	SSFG

Program Symbol	Equation Symbol
TAUKM1	$\Gamma_{k, k-1}$
TRFG	TRFG
TRNIB	TRNIB
UM1VEC	$\underline{u}_{k-1}$
UVECM1	$\underline{\hat{u}}_{k-1}$
WM1	$\underline{w}_{k-1}$
XDIF	$\underline{x}_{DIF}$
XKVEC	$\underline{x}_k$
XM1VEC	$\underline{x}_{k-1}$
XOVEC	$\underline{x}_o$
XSQGK	$\underline{\tilde{x}}_k$
YA	${}_i\underline{Y}_k$
YSTA	${}_i\underline{Y}^*{}_k$
ZETA	${}_o\zeta$
ZETZA	${}_i\zeta$

7.1.7 Guidance

Equation Symbol	Program Symbol	Equation Symbol	Program Symbol
$\underline{c}'$	CPM	Λ_c	CLAM
$\underline{c}''$	CDPM	Λ_{c+1}	LAMC1
D_Q	DQ	Π_c	PIC
		Π'_c	PIP
a^M	AM		
		$\Phi_{c,o}$	PHICO
$a^P{}_{c-1}$	APCM1	$a^{\Phi}{}_{c,c-1}$	APHI
		$\Phi_{c+1,o}$	PHIC10
Q_c	QC	$a^{\Phi}{}_{c+1,c}$	APHIC1
$_1Q_c$	QCI	$\Phi_{c+1,N}$	PHICIN
$_1Q_{oo}$	Q100		
T_Q	TQ		
$\underline{u}_c$	UMIVEC		
$\underline{\hat{u}}_c$	UVECM1		
$\underline{w}_c$	WM1		
$a\underline{\hat{x}}_c$	AXHC		
$\Gamma_{c,c-1}$	GMA		
$a^{\Gamma}{}_{c,c-1}$	AGMA		
$c^{\Gamma}{}_{c,c-1}$	CGMA		
$\Delta_{c,c-1}$	DLT		

Program Symbol	Equation Symbol	Program Symbol	Equation Symbol
AGMA	${}_a\Gamma_{c,c-1}$	TQ	T_Q
AM	${}_aM$		
APCM1	${}_aP_{c-1}$	UMIVEC	$\underline{u}_c$
APHI	${}_a\Phi_{c,c-1}$	UVECM1	$\underline{\hat{u}}_c$
APHIC1	${}_a\Phi_{c+1,c}$		
AXHC	${}_a\underline{x}_c$	WM1	$\underline{w}_c$
CDPM	$\underline{c}''$		
CGMA	${}_c\Gamma_{c,c-1}$		
CLAM	Λ_c		
CPM	$\underline{c}'$		
DLT	$\Delta_{c,c-1}$		
DQ	D_Q		
GMA	$\Gamma_{c,c-1}$		
LAMC1	Λ_{c+1}		
PHICO	$\Phi_{c,o}$		
PHIC10	$\Phi_{c+1,o}$		
PHICIN	$\Phi_{c+1,N}$		
PIC	Π_c		
PIP	Π'_c		
QC	Q_c		
QC1	${}_1Q_c$		
Q100	${}_1Q_{oo}$		

7.2 PROGRAM LISTING

The original of the program listing is supplied with the program decks.